PROJECT REPORT ON

INTEGRATED ONLINE TEACHER ASSISTANCE PORTAL USING MACHINE LEARNING TECHNIQUES

submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor Of Technology

in

Computer Science & Engineering

By

JAHNVI TYAGI (00601012016)
PRIYA GUPTA (01701012016)
MEHAR KAUR (06201012016)

Under the guidance of

Ms. Monika Choudhary
Assistant Professor

INDIRA GANDHI DELHI TECHNICAL UNIVERSITY FOR WOMEN
(Formerly Indira Gandhi Institute of Technology)
Kashmere Gate, Delhi-110006

May 2020

CERTIFICATE

This is to certify that the work presented in this B.Tech. Major Project titled **"Integrated Online Teachers Assistance Portal using Machine Learning Techniques: IGDTU_GUIDE"** is an authentic record of our own work under the supervision of "**Ms. Monika Choudhary,** Assistant Professor", "Department of Computer Science and Engineering".

It is submitted in partial fulfillment of the requirements for the award of the Bachelor of Technology in Computer Science & Engineering at Department of CSE, Indira Gandhi Delhi Technical University for Women.

Jahnvi Tyagi	**Priya Gupta**	**Mehar Kaur**
(00601012016)	**(01701012016)**	**(06201012016)**

This is to certify that this work has been done under my supervision and guidance. It has not been submitted elsewhere either in part or full, for award of any other degree or diploma to the best of my knowledge and belief.

Date:

Ms. Monika Choudhary

Assistant Professor

IGDTUW

ACKNOWLEDGEMENT

We would like to place on record our deep sense of gratitude to **Prof. Seeja K.R** , HOD-Department of Computer Science and Engineering, Indira Gandhi Delhi Technical University for Women Delhi, for her generous guidance, help and useful suggestions.

We express our sincere gratitude to **Ms Monika Choudhary**, Assistant Professor, for her stimulating guidance, continuous encouragement and supervision throughout the course of the project work.

We would also like to thank the entire staff of the Computer Science Department for their constant motivation, guidance and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

With the increasing use of technology in every sphere of the world, the education sector is also witnessing automation -:E-learning, chatbots, virtual classes, automated grading,Plagiarism check etc. With this project, we aim to build an online portal that will assist both professors and students to improve the learning experience and reduce the burden on professors. The model this paper proposes will consist of four separate modules - summary generation, question generation, Internal Plagiarism check and essay grading implemented using machine learning and deep learning techniques. To our knowledge, no such integrated platform exists till date.

Keywords: machine learning, deep learning, summary generation, question generation, essay grading, Internal Plagiarism detection

INTRODUCTION

Role of AI in education

Automated Grading: AI based grading systems are likely to help teachers and reduce their burden when it comes to online teaching. Even though online graders are extensively used for grading MCQ based answers, work is being done in essay grading type answers too.

Helpful for Teachers: AI chatbots can be made available on the portal to answer "frequently/most asked questions" to reduce the dependence of learning on teachers.

Helpful for Students:AI systems can learn from past data to make predictions for the future. Given a student's school and college data, AI can identify their strengths and weaknesses and improve their learning strategy.

Cater to Student Needs: AI can be designed to overcome physical disabilities in students eg. visually impaired students can make use of the Text-to-speech feature to listen to the content in various PDFs.

Help weak students with studies: Artificial Intelligence will benefit all the students, including the weaker ones. Students who miss out on questions and lag behind will be identified easily by observing patterns and made into teacher's notice.

Plagiarism on the rise

A form of cheating, Plagiarism is morally and ethically repugnant and is intellectually deceitful. It can be defined as being *"Theft or misappropriation of intellectual property and the substantial unattributed textual copying of another's work."*

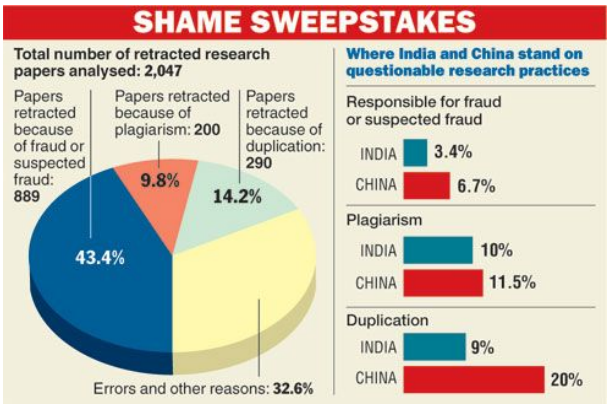


Fig 1. Statistics on Plagiarism[1]

India has witnessed an increased number of academic plagiarism cases which can be attributed to increased pressure to publish, lack of training in ethical scientific writing,lack of will, oversight and lack of statutory controls and clear policies to deal with scientific misconduct. Increased plagiarism in the higher education sector of India will hamper growth and block innovation.

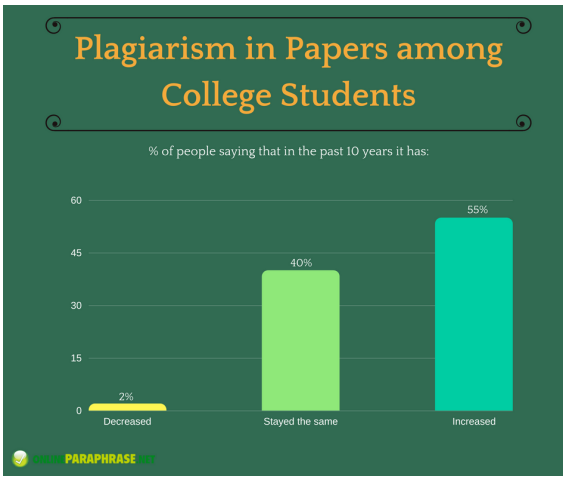


Fig 2. Plagiarism among college students[2]

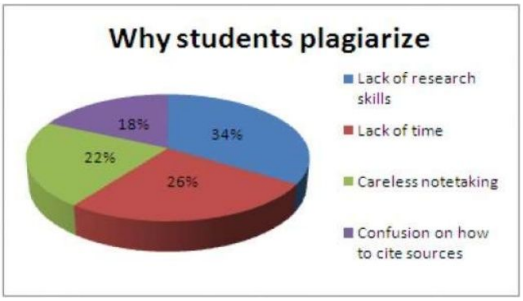


Fig 3. Why Students plagiarize[3]

Usually journal editors are the first to detect Plagiarism But they lack resources, expertise and more importantly authority to conduct the confirmatory investigations. Investigating research misconduct is not only tough but time consuming necessitating scientific, administrative, legal expertise and the will to act.

Literature survey:

Title & Author	Publication	Dataset	Methodology proposed	Metric used
Plagiarism detection using Semantic Knowledge graphs by Kunal Khadilkar, Poojarani Bone[1]	2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)	Not specified	This paper proposes a new method to detect plagiarism when the structure of the sentence is changed using semantic knowledge graphs. The method uses Named Entity Recognition as well as semantic similarity between sentences to detect possible cases of plagiarism. The doubtful cases are visualized using semantic Knowledge Graphs for thorough analysis of authenticity.	Not mentioned
Machine Learning models for paraphrase identification and its application on Plagiarism detection by Ethan Hunt, Ritvik Janamsetty,	2019 IEEE International Conference on Big Knowledge (ICBK)	Not specified	Various Machine Learning algorithms have been implemented such as Logistic Regression, SVM and Neural Networks for paraphrase identification which is further extended to plagiarism detection. The best suited model was found to be Recurrent Neural Networks (RNN).	Not mentioned

Chanana Kinares et al				
Internal Plagiarism Detection Using Machine Learning-Based Paraphrase Recognizer by A. Chitra and Anupriya Rajkumar[3]	Journal of Intelligent Systems 25.3 (2016): 351-359.	Microsoft Research Paraphrase Corpus (MRPC)	A SVM-based paraphrase recognizer is used to extract various lexical, syntactic, and semantic features and then GA is used to identify the best features. Then the SVM classifier labels the sentence pairs as positive or negative. The decisions obtained for individual sentence pairs are then aggregated and if the percentage exceeds a given threshold, the passage is declared to be plagiarized.	Not mentioned
NLP Based Text Summarization Using Semantic Analysis by Hamza Shabbir Moiyadi, Harsh Desai, Dhairya Pawar ,Geet Agrawal, Nilesh M.Patil[4]	International Journal of Advanced Engineering, Management and Science (IJAEMS) [Vol-2, Issue-10, Oct- 2016]	Latent Semantic Analysis, Singular Value Decomposition (SVD)	Deep Understanding is performed using Sentence Extraction, Paragraph Extraction and Machine Learning techniques are applied.	Not mentioned
Automatic Extractive Text Summarization Using K-Means Clustering by M R Prathima, H R Divakar[5]	International Journal of Computer Sciences and Engineering Vol.-6, Issue-6, June 2018	Extractive Summarization, Natural language Processing (NLP), Clustering, Support-VectorMachine (SVM), Tokens.	The term frequency-inverse document frequency (tf-idf) is used which is incredibly powerful and which is used to judge the topic of an article by the words that it contains. When a set of data points is given then, the clustering algorithm can be used to classify each data point into a particular class. An algorithm will be generated that contains clustering machine learning technique i.e., Support-Vector-Machine (SVM).	Rouge1, Rouge2 scores
Deep Learning in Automatic Text Summarization by Som Gupta and S.K Gupta[6]	International Journal of Computer Science and Information Security (IJCSIS), Vol. 16, No. 11, November 2018	Sentence Ranking and Sentence Selection are the two most important tasks which are used for creating extractive summaries. For abstractive summaries, we used bag of words based encoder and convolutional	Restricted Boltzmann Machine, Sequence2Sequence Models using Encoder-Decoder approach and Unsupervised approaches have been used for summarization purpose.	Not mentioned

		neural networks along with the attention mechanism and beam-search based decoder to create the summary.		
Automatic Question Generation from Text by Himanshu Jethwani, Mohd Shahid, Husain Mohd Akbar[7]	International Journal for Innovations in Engineering, Science and Management Volume 3, Issue 4, April 2015	The basic approach to solve the problem is integration and conversion. First the sentence/s should be broken into parts and classified and then it should be converted into questions. The basic approach for extracting simple statements to generate question is as follows Here we have Input: complex sentence/s Output: set of simple declarative sentences	Our method: 1. Uses rules to extract and simplify sentences 2. Is motivated by linguistic knowledge 3. Outperformed a sentence compression baseline. Our idea here is to extract and simplify multiple statements from complex sentences including operations for various syntactic constructions encoded with pattern matching rules for trees.	Not mentioned
A Memory Augmented Neural Model for Automated Grading by Siyuan Zhao, Yaqiong Zhang ,Xiaolu Xiong ,Anthony Botelho ,Neil Heffernan[8]	Proceedings of the Fourth (2017) ACM Conference on Learning@ Scale. 2017.	Kaggle Automated Student Assessment Prize (ASAP) competition.	A simple multi-layer forward neural networks (FNN) model having the layers: input representation layer, memory addressing layer, memory reading layer, and output layer, was trained and used for prediction.	Quadratic weighted Kappa (QWK)

Task-Independent Features for Automated Essay Grading by Torsten Zesch ,Michael Wojatzki, Dirk Scholten-Akoun	Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications. 2015.	ASAP Essay grading challenge	Take a set of essays, then after preprocessing, perform syntactic parsing and extracts a list of features to be given as input to the ML Model.	Quadratic web kappa (QWK)
A Neural Approach to Automated Essay Scoring by Kaveh Taghipour and Hwee Tou Ng[10]	Proceedings of the 2016 conference on empirical methods in natural language processing. 2016.	Kaggle Automated Student Assessment Prize (ASAP) competition.	Several neural networks like LSTM, CNN, RNN, Attention networks, were trained and RMSProp optimizer was used.	Quadratic weighted Kappa (QWK), Comparison with EASE(Enhanced AI Scoring Engine)

Table 1: Literature survey

Value Proposition:

Our Problem definition is as follows:
"Build an online portal to assist teachers and students in improving the classroom learning experience: automatic summary and question generation, detect level of internal Plagiarism and grade assignments, using Artificial Intelligence techniques"
We chose this problem statement with the following aim:

1. **Improved Classroom Learning Experience:** Classroom learnings would become more interactive and less focussed on scribbling notes ensuring optimum use of class time. The teacher would *upload a PDF version of the notes* on the portal after every class and the model would *automatically generate a short summary and question bank* for students to refer to and practice.
2. **Improved Understanding of Topics:** *Automatic grading using algorithms and Internal Plagiarism check of assignments* would help teachers identify students with poor understanding of the topic and work with them on their weak areas.
3. **Efficient Revision Tool:** The model would remove unnecessary texts, images and paragraphs while generating summary of the notes. This would *enable quicker and more efficient last-minute revisions for students.*

IGDTUW_Guide: Help fight the Covid Battle

In today's times, when the whole world is fighting a global pandemic, COVID'19, the education sector has taken a hit and there has been an urgent need to shift to online portals for classes as well as exams. We propose this portal to ease this process of online learning and aid teachers in grading the assignments quicker. It also bridges the gap between teachers and students as all of

the faculty's content will be updated on a centralized portal with corresponding summary and question generated for better learning for the students.

REQUIREMENTS SPECIFICATION

Project Development Tools :

Fig 4. Tools and techniques

Module	Tools and Techniques
User Interface	HTML, CSS, Javascript
Backend Web development	Flask
Web Hosting	Heroku
Cloud Storage	Dropbox
Machine Learning Models using Python3	
Summary Generation	Text Rank Algorithm, LSA, TF-IDF
Internal Plagiarism detection	Sequence Matcher, Vector Space Model
Question Generation	Using NLTK library
Essay Grading	SVM, Feed Forward Neural Network

Table 2. Tools used

SYSTEM DESIGN

User Interface Flowchart - Teachers

After logging in the website, the teachers dashboard will have 4 features - *Upload new content* (PDF) via a form, view the previously uploaded content and *generate corresponding questions and summary of the PDF*, upload a long essay- type question for "Internal Assessment" of students & finally view the answers submitted by students roll-number wise and detect internal plagiarism in the answers and view machine generated grade.

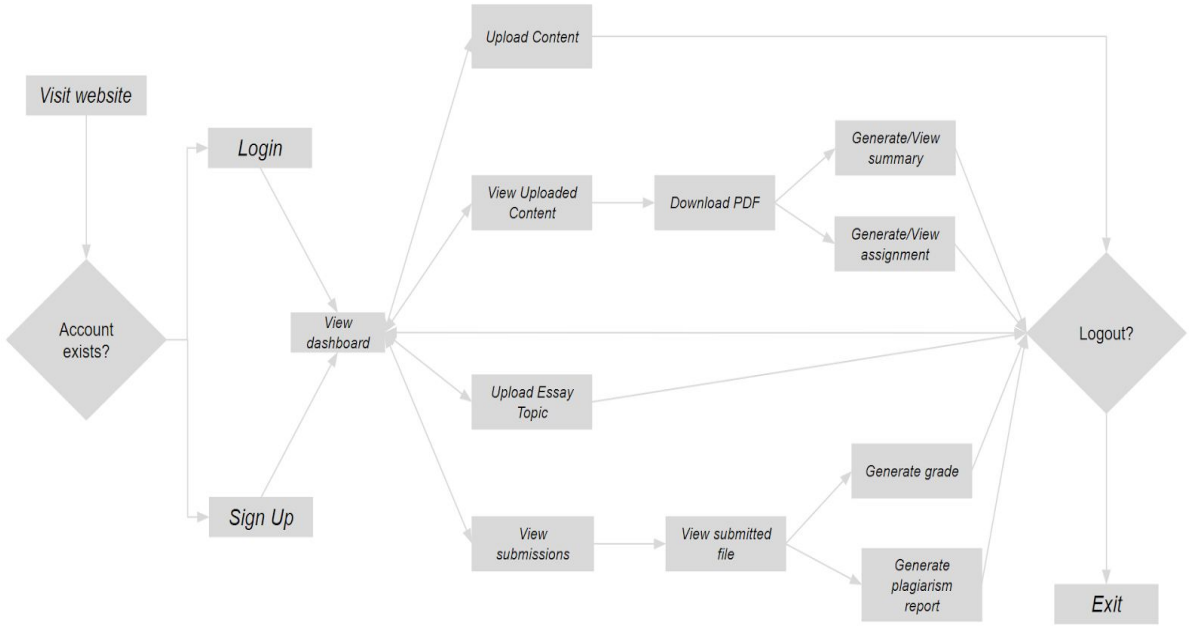


Fig 5. UI flowchart for Teachers

User Interface Flowchart - Student

After logging in, a student would be available to navigate through every faculty member's page in the CSE department and view the notes uploaded by them. The portals provide functionalities to view original PDF, summarised PDF, question bank and upload their answers.

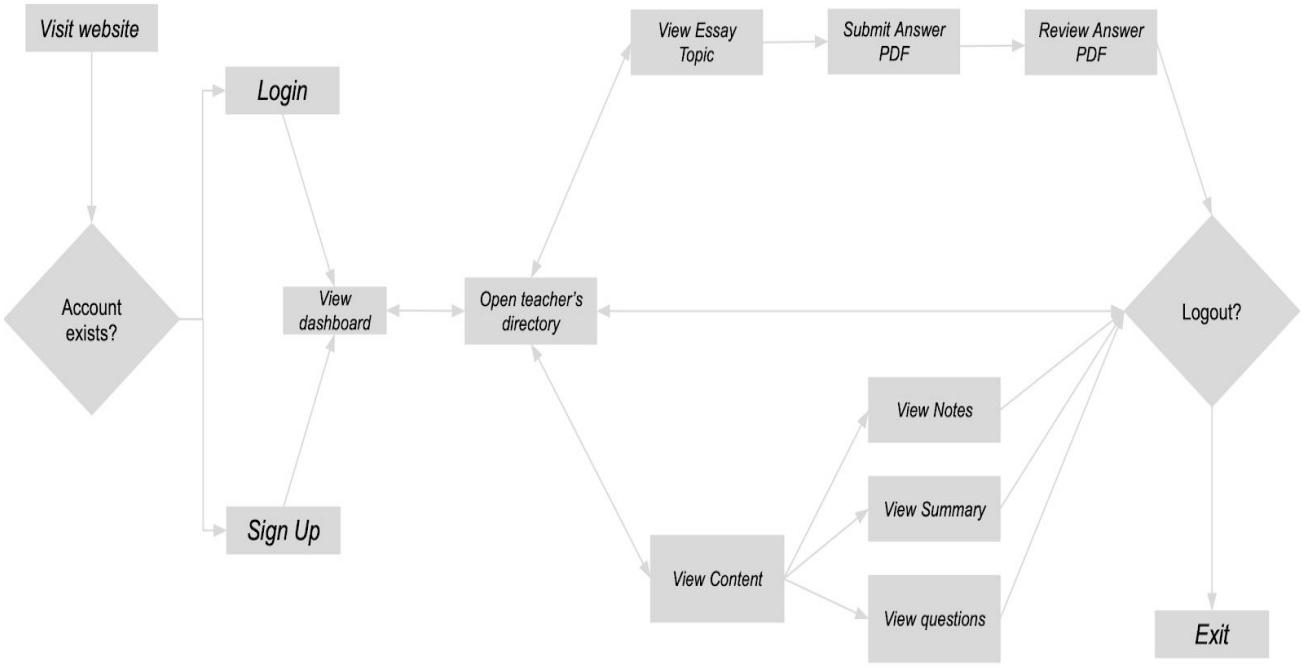


Fig 6. UI flowchart for Students

WEB PORTAL:

1. INTERFACE FOR TEACHERS

- **Upload notes for topics taught in class**
  Teachers can upload notes in PDF format for minor-1, minor-2 and major.

Fig 7. Teachers dashboard - Upload your notes

- **Upload essay topic for internal grading**
  Upload essay topic along with word limit and submission date which would be evaluated by the Grading and Internal Plagiarism machine learning models.

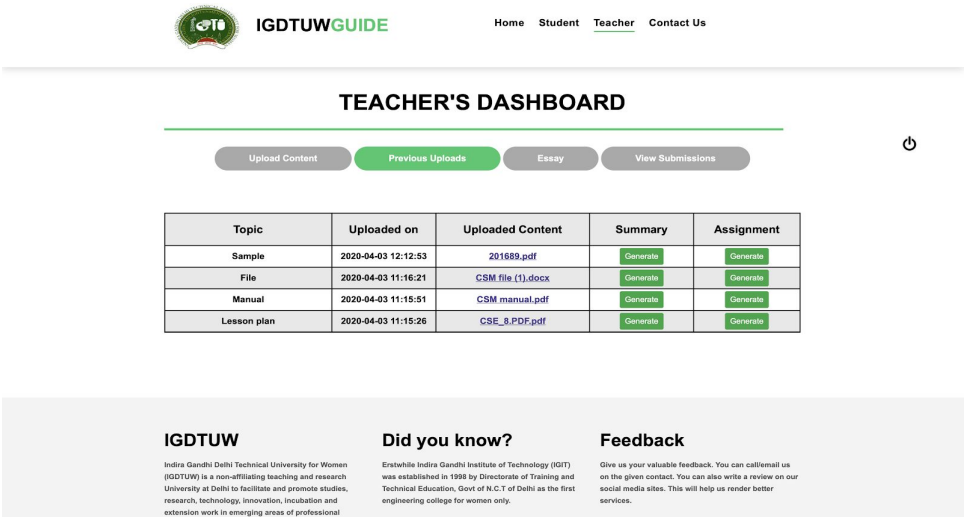


Fig 8. Teachers dashboard - Upload essay topic

- **Keep track of previously uploaded notes**

On clicking this tab, teachers can easily view a complete list of topic-wise uploaded notes along with the date of uploading it. Additionally, if the prof. clicks on generate summary/ assignment, the generated documents would be available on the students dashboard for quick revision.

Fig 9. Teachers dashboard - View previous uploads

- **View students submissions**

  On clicking view submissions, a roll number wise list of all submitted answers would be available on the teachers dashboard. The prof. Can also check the amount of intrinsic Internal Plagiarism in the answers and view grades assigned by our ML model based on certain features.

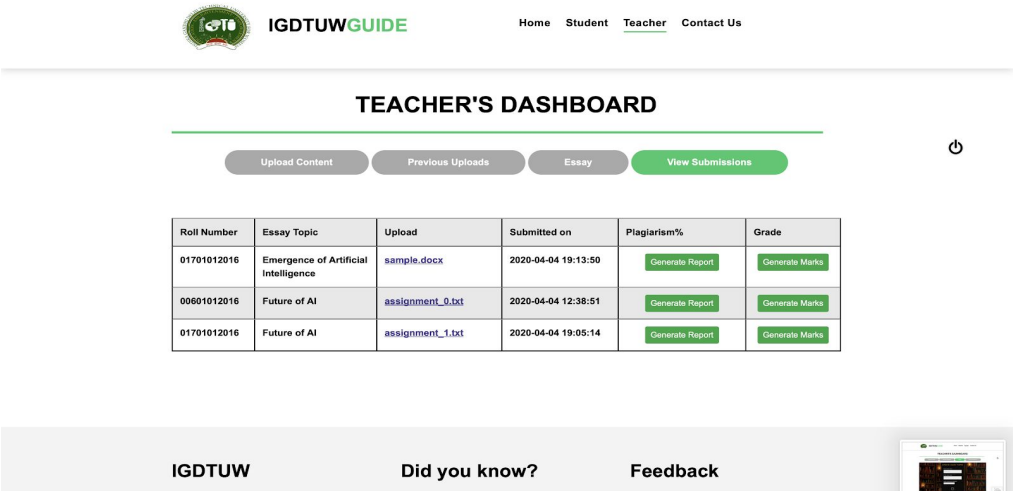


Fig 10. Teachers dashboard - View submissions

2. INTERFACE FOR STUDENTS

- **Availability of every teacher's content**
  Every faculty member in the CSE department has a separate directory to upload and store their notes which can be accessed by the students 24x7.

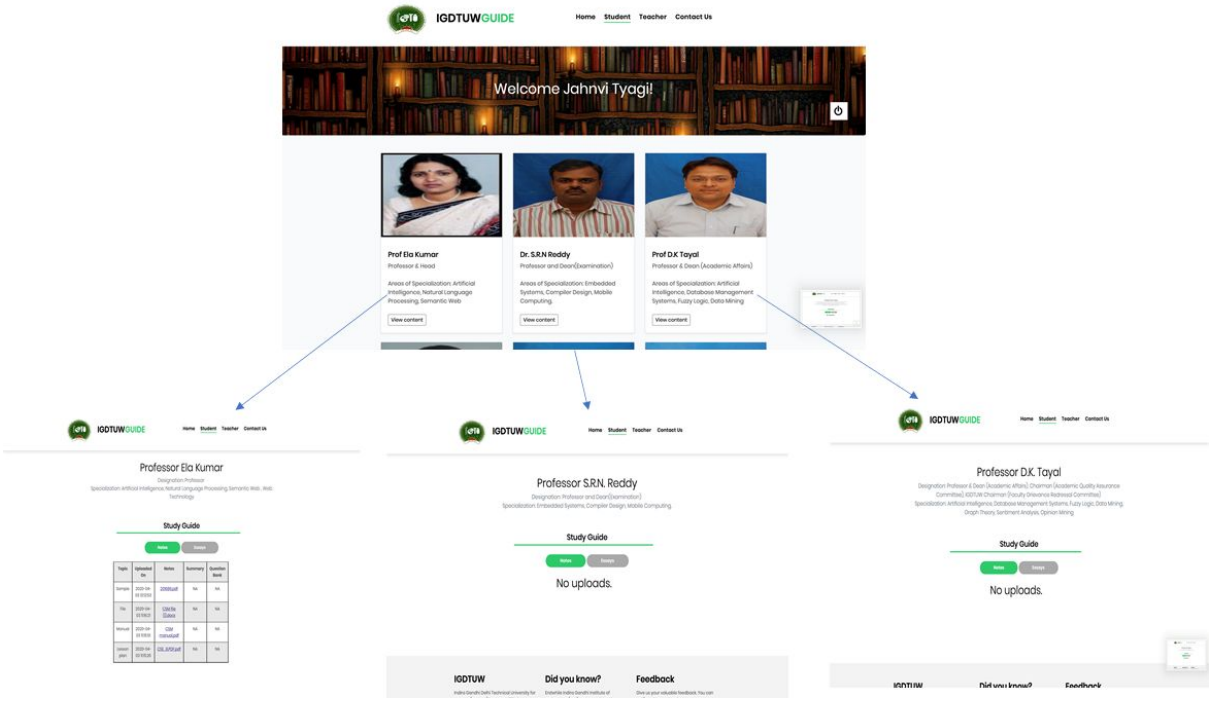
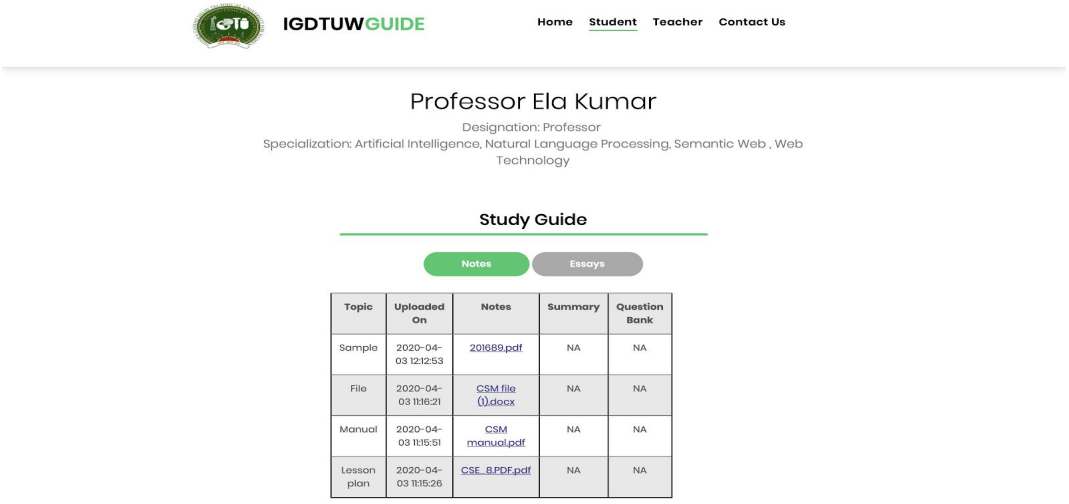


Fig 11. Student dashboard - Teacher's content

- **Access complete classroom notes**
  In the notes section of the study guide, students can view and access full version notes, their automatic generated summary or question bank for quick revision before the exams.



Topic	Uploaded On	Notes	Summary	Question Bank
Sample	2020-04-03 12:12:53	201689.pdf	NA	NA
File	2020-04-03 11:16:21	CSM file (1).docx	NA	NA
Manual	2020-04-03 11:15:51	CSM manual.pdf	NA	NA
Lesson plan	2020-04-03 11:15:26	CSE_B.PDF.pdf	NA	NA

Fig 12. Student dashboard - Classroom notes

- **Upload Essay Answers**

Students can upload their essay answers online in PDF format before the due-date specified by the professor.

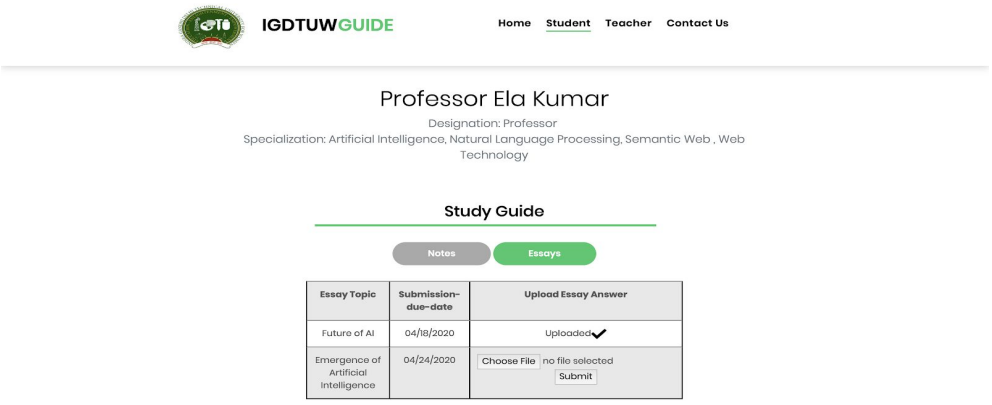


Fig 13. Student dashboard - Submit essay

AUTOMATED SUMMARY GENERATION:

TECHNIQUES

If Cristiano Ronaldo didn't exist, would Lionel Messi have to invent him?

The question of how much these two other-worldly players inspire each other is an interesting one, and it's tempting to imagine Messi sitting at home on Tuesday night, watching Ronaldo destroying Atletico, angrily glaring at the TV screen and growling: "Right, I'll show him!"

As appealing as that picture might be, however, it is probably a false one — from Messi's perspective, at least.

He might show it in a different way, but Messi is just as competitive as Ronaldo. Rather than goals and personal glory, however, the Argentine's personal drug is trophies.

Ronaldo, it can be said, never looks happy on the field of play unless he's just scored a goal — and even then he's not happy for long, because he just wants to score another one. And that relentless obsession with finding the back of the net has undoubtedly played a major role in his stunning career achievements.

Messi, though, is a different animal, shown by the generosity with which he sets up team-mates even if he has a chance to shoot, regularly hands over penalty-taking duties to others and invariably celebrates a goal by turning straight to the player who passed him the ball with an appreciative smile.

Rather than being a better player than Ronaldo, Messi's main motivations — according to the people who are close to him — are being the best possible version of Lionel Messi, and winning as many trophies as possible.

That theory was supported by Leicester boss Brendan Rodgers when I interviewed him for a book I recently wrote about Messi.

Figure 14 - Input text for summary generation analysis [4]

Consider the above article. It has been summarized using the given ***three extractive techniques*** and the best one has been chosen for further analysis:

1. LATENT SEMANTIC ANALYSIS

1.1 EXPLANATION

It is an extractive summarization algorithm using Algebraic - Statistical methods. LSA extracts hidden semantic structures, in other words it extracts the features which aren't clearly expressed or laid out in the words and sentences of the paragraph. These features are important to data , but are not rudimentary features of the dataset. It uses the concepts of Natural Language Processing and is an unsupervised Machine Learning algorithm.

1.2 WHY WE CHOSE THIS ALGORITHM

- LSA is a global algorithm and its scope is greater than other local based algorithms. It identifies the relations and patterns from all words, sentences and paragraphs in the document which may not be apparent to other local algorithms.
- It has minimum noise as compared to other algorithms and contains the most information. It also relates to lower dimensions which is perfect for the analysis of new algorithms.

1.3 OUR IMPLEMENTATION

Step 1: Clean the text using NLTK libraries
Step 2: Tokenize the text
Step 3: Convert the paragraphs into sentences (Sentence Splitter)
Step 4: Stop Word removal
Step 5: POS tagging for every sentence
Step 6: Find the stem and root of the word
Step 7: Matrix representation of the sentences
Step 8: Perform Singular Value Decomposition
Step 9: Extracting the top sentences

```
Do Messi and Ronaldo inspire each other? Ronaldo, it can be said,
never looks happy on the field of play unless he has / he is just
scored a goal - and even then he has / he is not happy for long,
because he just wants to score another one. Rather than being a
better player than Ronaldo, Messi\'s main motivations - according to
the people who are close to him - are being the best possible
version of Lionel Messi, and winning as many trophies as possible.
That theory was supported by Leicester boss Brendan Rodgers when I
interviewed him for a book I recently wrote about Messi. With the
very elite performers, that drive comes from within." "But I think
both those players inherently have that hunger to be the best
players they can be.
```

Figure 15 - Summary given by LSA

2. TEXTRANK ALGORITHM

2.1 EXPLANATION

TextRank is an unsupervised machine learning technique and is based on the extractive summarization mechanism.

2.2 WHY WE CHOSE THIS ALGORITHM

1. TextRank has actually been inspired by the PageRank algorithm which is widely used in the Google search engine to link web pages to one another.
2. PageRank is very efficient when it comes to going from one web page to another as these pages are linked to each other with certain probabilities also known as transitional probabilities.
3. More important pages have high values and vice versa.
4. TextRank uses sentences instead of web pages and various sentences are ranked according to their probabilities.
5. It gives immensely good results just like PageRank and hence we chose to go forward with this algorithm.

2.3 OUR IMPLEMENTATION

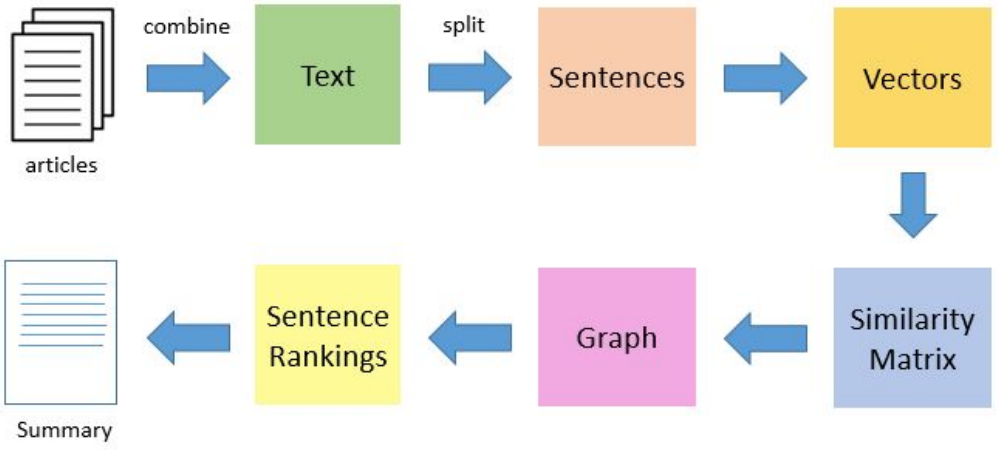


Figure 16: Flowchart for TextRank algorithm[5]

- The text present in the article in put together in the first step (concatenation)
- In the second step, the paragraphs or the text is divided into sentences.
- The next step involves finding the word embeddings or in other words, vector representation of each and every sentence.
- A similarity matrix is then constructed that contains the importance or the transition probabilities of various sentences.
- Taking the similarity score as edges and sentences as the vertices, a graph is created using the similarity matrix constructed in the previous step for calculating the sentence rank.

- In the last step, the summary is generated with a pre-defined number of sentences either manually entered by the user or chosen by the algorithm using the top-ranked sentences.

"He might show it in a different way, but Messi is just as competitive as Ronaldo. Ronaldo, it can be said, never looks happy on the field of play unless he has / he is just scored a goal - and even then he has / he is not happy for long, because he just wants to score another one. Rather than being a better player than Ronaldo, Messi's main motivations - according to the people who are close tohim - are being the best possible version of Lionel Messi, and winning as many trophies as possible. Do Messi and Ronaldo inspire each other? Messi and Ronaldo ferociously competing with each other for everyone else's acclaim is a nice story for fans to debate and the media to spread, but it has / it is probably not particularly true."

Figure 17 - Summary given by TextRank algorithm

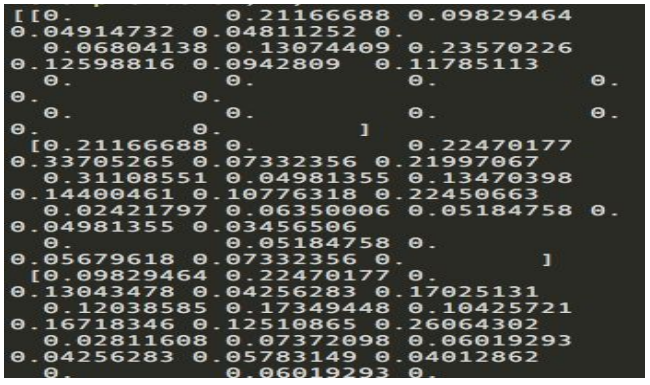


Fig 18.1 - Similarity matrix

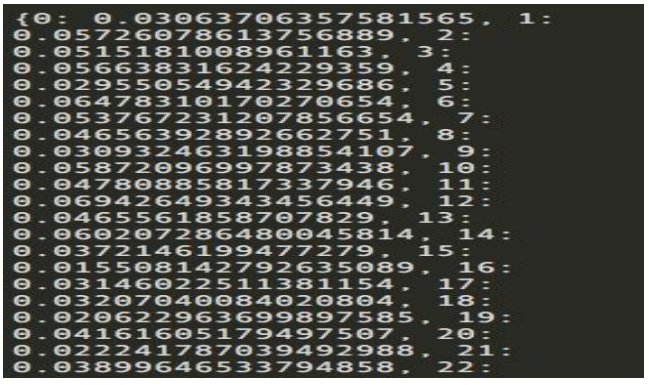


Fig 18.2 - Graph embedding

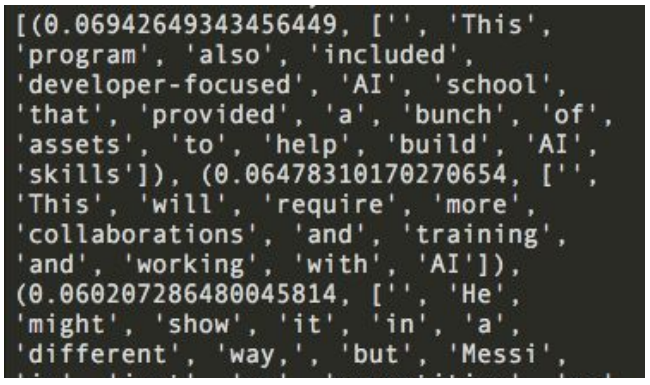


Fig 18.3 - Sentence rankings

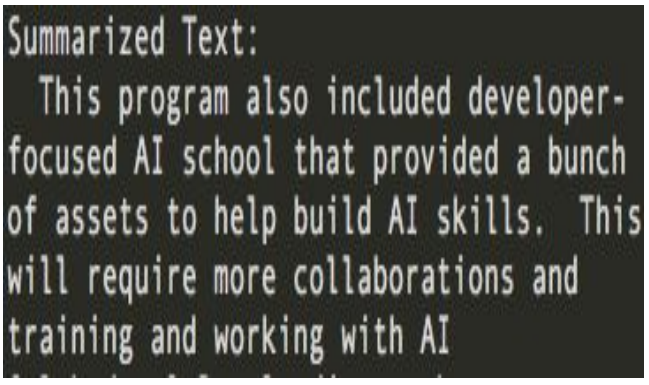


Fig 18.4 - Generated Summary

Figure 18 - Steps in TextRank algorithm

3. TF-IDF

3.1 EXPLANATION

TF IDF which is the short form of “Term Frequency–Inverse Document Frequency”, is a numerical metric which is used to quantify the degree of value of a term(word) in a document based on its frequency in the document or a set of documents at hand. The main idea behind the algorithm is as follows: If a word appears iteratively or many times in one documents then perhaps it is a keyword and must be important, therefore it is given a higher rank whereas if a word occurs multiple times in all other documents as well then it must be given a lower score as it may be a common English word (article, preposition, etc).Formula for calculating tf and idf:

TF(w) = (Frequency of term w appears in a document) / (Frequency of all the terms in the document)

IDF(w) = log_e(Total number of documents / Total occurence of documents with term w prevalent in it)

Hence **tf-idf** for a word is derived as: **TFIDF(w) = TF(w) * IDF(w)**

3.2 WHY WE CHOSE THIS ALGORITHM

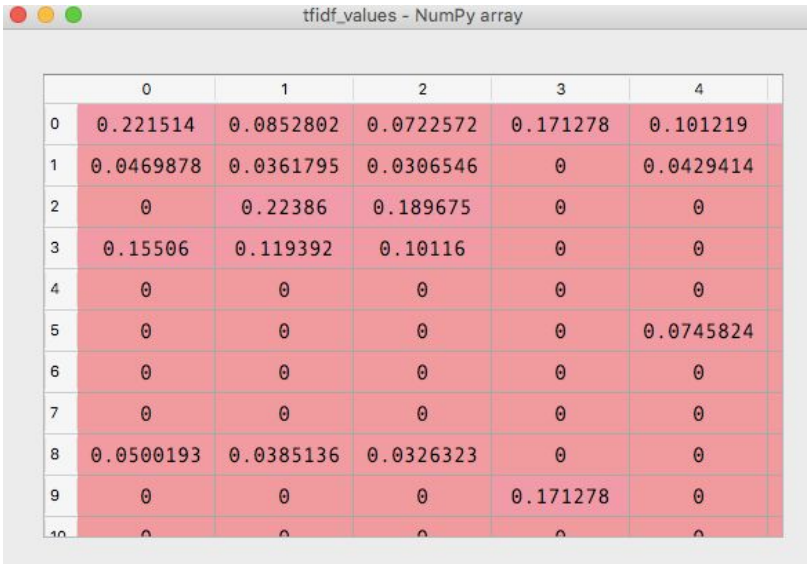
1. TF-IDF is an ideal choice for implementing summary generation algorithms as its encoding is not very demanding and hence can be corporated in other complex query retrieval algorithms as well.
2. Similarity between two documents can also be easily computed.
3. It is easy to compute and has a basic metric to derive the descriptive document.

3.3 OUR IMPLEMENTATION (STEPS)

1. Importing necessary libraries and initializing WordNetLemmatizer
2. Text preprocessing
3. Next step involves counting the number of times each word occurs in the document in order to calculate its importance.
4. Calculating sentence score
   - 4.1. POS tagging function
   - 4.2. Word tf idf function
   - 4.3. tf score function
   - 4.4. idf score function
   - 4.5. tf idf score function
5. Finding most important sentences

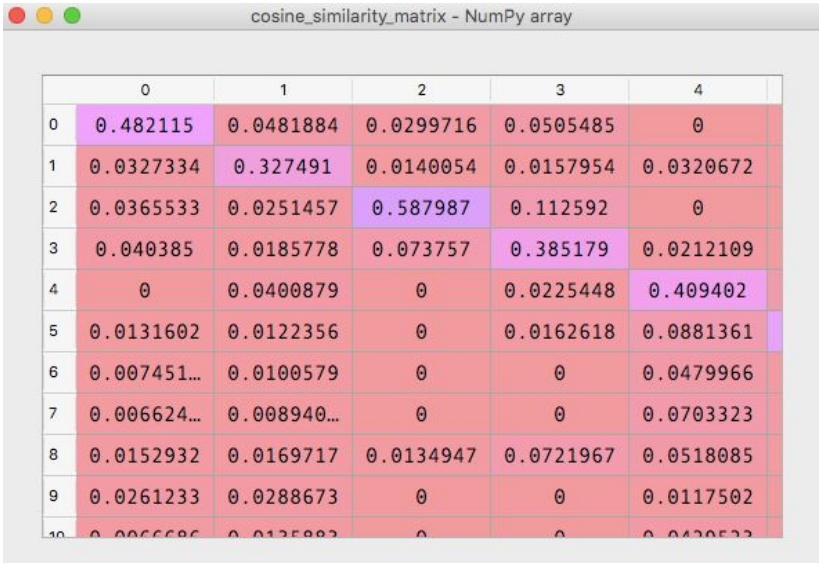
ide	Type	Size	Value
0	str	1	an
1	str	1	oper
2	str	1	system
3	str	1	os
4	str	1	is
5	str	1	interfac
6	str	1	between
7	str	1	a
8	str	1	comput
9	str	1	user

Figure 19 - Unique words generated by tf-idf algorithm



	0	1	2	3	4
0	0.221514	0.0852802	0.0722572	0.171278	0.101219
1	0.0469878	0.0361795	0.0306546	0	0.0429414
2	0	0.22386	0.189675	0	0
3	0.15506	0.119392	0.10116	0	0
4	0	0	0	0	0
5	0	0	0	0	0.0745824
6	0	0	0	0	0
7	0	0	0	0	0
8	0.0500193	0.0385136	0.0326323	0	0
9	0	0	0	0.171278	0

Figure 20 - tfidf values



	0	1	2	3	4
0	0.482115	0.0481884	0.0299716	0.0505485	0
1	0.0327334	0.327491	0.0140054	0.0157954	0.0320672
2	0.0365533	0.0251457	0.587987	0.112592	0
3	0.040385	0.0185778	0.073757	0.385179	0.0212109
4	0	0.0400879	0	0.0225448	0.409402
5	0.0131602	0.0122356	0	0.0162618	0.0881361
6	0.007451…	0.0100579	0	0	0.0479966
7	0.006624…	0.008940…	0	0	0.0703323
8	0.0152932	0.0169717	0.0134947	0.0721967	0.0518085
9	0.0261233	0.0288673	0	0	0.0117502

Figure 21 - Cosine Similarity Matrix

AUTOMATED QUESTION GENERATION:

1. EXPLANATION

A text file is passed as an argument to the program. The text file is read using a Python package called textblob. Each paragraph is further broken down into sentences using the function parse(string): And each sentence is passed as string to function AQGenerator().

2. WHY WE CHOSE THIS TECHNIQUE

This algorithm focuses on the generation of different types of questions such as What, When, Whom etc. It does so by making use of powerful NLP techniques such as POS tagging, Word Sense Disambiguation (WSD) and Named Entity Recognition. Semantic

relations between the words in a sentence are also preserved and the frequency of questions produced increases as the length of the paragraph.

OUR IMPLEMENTATION

A. Sentence Selection

This phase is an extremely crucial stage in the algorithm implementation as here we select the sentences from which questions would be generated in the next stage. It consists of two sub phases:

- Paragraph processing
- Feature Selection

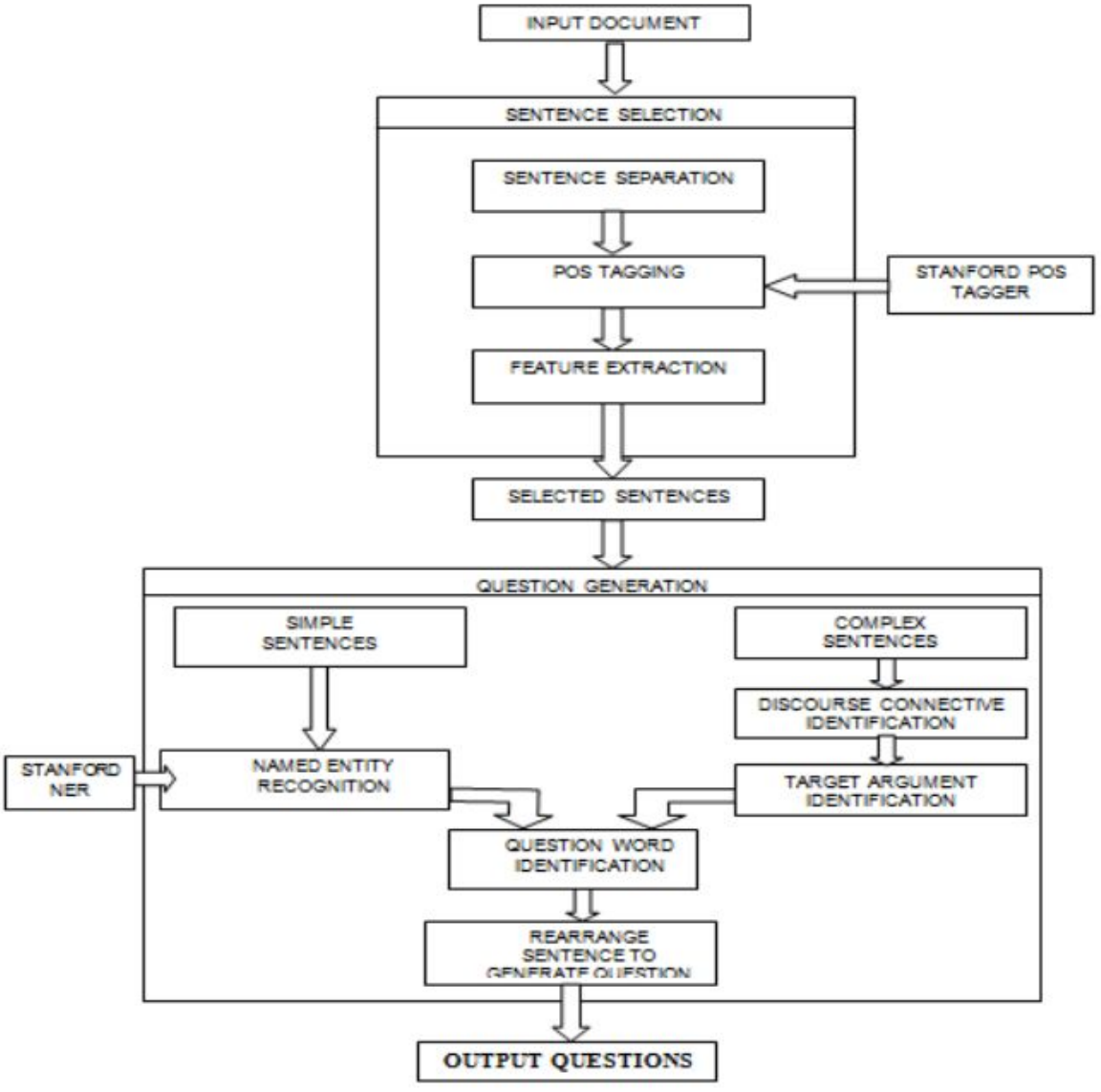


Figure 22 - Flowchart for Question Generation process[6]

After this phase is finished, we get a set of selected sentences which will be used in the second step of question generation.

1) Paragraph Processing

This phase consists of scanning the input paragraph from which questions are to be generated. The paragraph based on full stops is split into individual sentences. Then these split up sentences are processed by a POS Tagger which outputs the sentence along with the POS tags of each and every word. This is then utilised further in the feature extraction and automatic question generation process.

2) Feature Selection

1. **First Sentence**

   It has been observed that the first sentence of any documents generalises a summary of the entire document and hence is important to be considered as a feature. This feature tells us if a particular sentence is the first sentence of a document or not. It also gives us an idea about the question of general scope.

2. **Last Sentence**
   Unlike the first sentence, this feature would tell us if a particular sentence is the last sentence of a document or not. Generally, the last sentence provides a comprehensive conclusion of the document. Thus, extracting this feature gives us a general idea about the concluding words in a particular document.

3. **Length**
   Length of the sentences plays a major role in the purpose of question generation. This feature simply tells us what the length of a sentence is, in other words the exact number of tokens/words present in it. Usually, a short sentence doesn't prove to be a good choice for the process of question generation. We define a term 'n' as the minimum number of tokens a sentence must have in order to be chosen as the basis to form questions.

4. **Common Tokens**
   Usually, a sentence is considered important if it has the words which are also contained by the titles and subtitles of the paragraphs and documents and proves to be a vital choice for question generation. Thus, this feature counts the common words, usually nouns and adjectives that the titles and sentences have in common.

5. **Frequency of nouns**
   The frequency of nouns (NN, NNP, NNS, NNPS) in a sentence is provided by the POS Tagger in the first phase of this algorithm. As the count of nouns in a sentence increases so does its informational value. Hence, a sentence with more nouns provides more information and is chosen for further analysis.

6. **Frequency of pronouns**
   Again, the frequency of pronouns (PRP) in a sentence is provided by the POS Tagger. As the count of pronouns increases, the informational value of that sentence

decreases and it is not chosen for further analysis as it doesn't usually provide a basis for forming fruitful questions.

7. **Discourse Connective**

The prevalence of discourse connectives in a sentence is judged by this feature. Discourse connectives prove to be an important aspect in making the text coherent and can be easily utilised in the formation of wh-type questions.

B. Question Generation

1. **Generating questions on simple sentences**

The simple sentences are divided into subsections of any English sentence i.e Subject, Verb, Object. Then to obtain the coarse class classification of the subject and object a Named Entity Recognizer (NER) is passed over it. It then gives the appropriate tags their values such as Human, Time, Entity etc. The classification is as follows:

Human: The name of a person is denoted by this.

Entity: This includes any object ranging from mountains to flora and fauna.

Time: This denotes any time, day, month or year or a certain period such as Friday, 11pm, 2018 etc.

Location: It tells the type of organization, city, country or place of the tagged identity.

Count: This contains all those elements which are tangible and can be counted such as 5 apples, 7 women, 3 boxes, etc. It also takes into account weights and sizes of objects.

Organization: Organizations usually comprise of institutes, governmental structures, market, companies, establishments, etc.

We now take into account the relationship between the words in the sentence once they have been classified into coarse classes. For instance, if the sentence has the structure "Human Verb Human", then it will be classified as "whom and who" question type and if it is followed by a preposition which denotes date, then we conjunct the "When" question type to its classification.

2. **Generating questions on complex sentences**

The sentences that comprise discourse connectives i.e conjunctions like because, for example, since, when etc. are generally termed as complex sentences.

Discourse Connective	***Question type***
Because	*Why*
Since	*When, Why*
When	*When*
As a result	*Why*
For example	*Give an example where*
For instance	*Give an instance where*

Figure 23 - Discourse Connectives and Q-type

```
NNS     Noun, plural
JJ      Adjective
NNP     Proper noun, singular
VBG     Verb, gerund or present participle
VBN     Verb, past participle
VBZ     Verb, 3rd person singular present
VBD     Verb, past tense
IN              Preposition or subordinating conjunction
PRP     Personal pronoun
NN      Noun, singular or mass
```

Figure 24 - Part-of-speech tags used[7]

```
l1 = ['NNP', 'VBG', 'VBZ', 'IN']
l2 = ['NNP', 'VBG', 'VBZ']
l3 = ['PRP', 'VBG', 'VBZ', 'IN']
l4 = ['PRP', 'VBG', 'VBZ']
l5 = ['PRP', 'VBG', 'VBD']
l6 = ['NNP', 'VBG', 'VBD']
l7 = ['NN', 'VBG', 'VBZ']
l8 = ['NNP', 'VBZ', 'JJ']
l9 = ['NNP', 'VBZ', 'NN']
l10 = ['NNP', 'VBZ']
l11 = ['PRP', 'VBZ']
l12 = ['NNP', 'NN', 'IN']
l13 = ['NN', 'VBZ']
```

Figure 25 - Combination of POS tags used[8]

Each sentence is parsed using English grammar rules with the use of condition statements. A dictionary is created called bucket and the part-of-speech tags are added to it. The sentence which gets parsed successfully generates a question sentence. The generated question list is printed as output.

EXAMPLE: Input Text

"My best friend and I have been studying in the same school since kindergarten. We have been classmates each year at school. We share a very close bond and have a special friendship that we cherish and treasure. My friend is my partner, sitting beside me in class. She is kindly and helpful, and if I have any difficulties in understanding any topic in my studies, or in completing my homework or school project, she helps me. She is brilliant in mathematics and the sciences, while I am good at English. So we both help each other in whatever way possible. She helps me without ever belittling me. I greatly appreciate the quality in her. She does not make me feel obliged."

Figure 26 - Example given to the algorithm for question generation process [9]

Output:-

```
Q-01: Have you been classmates each year?
Q-02: Have you been at school?
Q-03: Who have been classmates each year at school?
Q-04: Who cherish and treasure?
Q-05: Who helps me?
Q-06: Who is good at English?
Q-07: Who helps me without ever belittling me?
Q-08: Who ever belittling me?
Q-09: Who greatly appreciate the quality in her?
Q-10: Whom she does not make feel obliged?
```

Figure 27 - The result of question generation process[10]

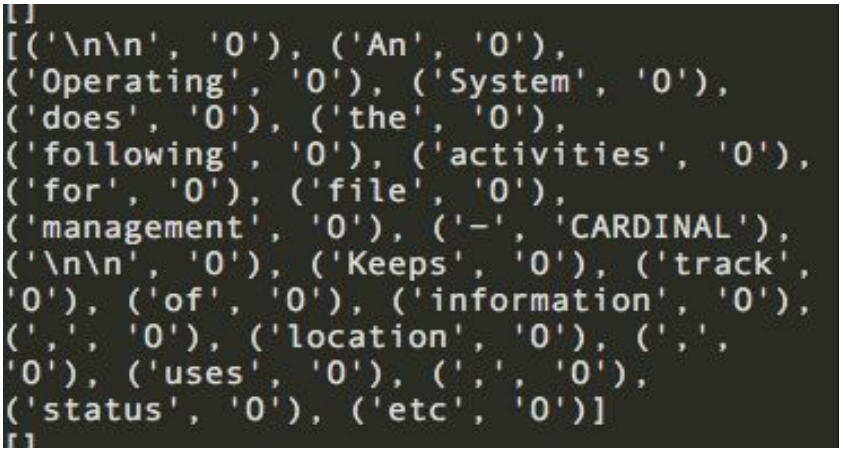


Figure 28.1 - Tagged words

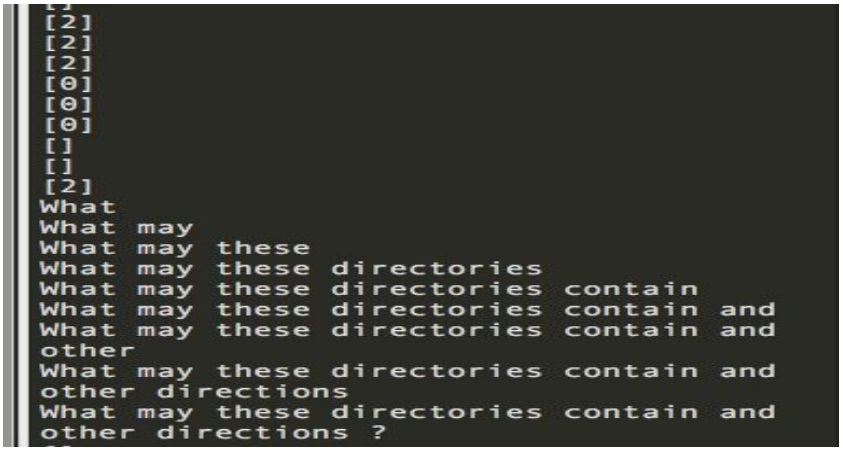


Figure 28.2 - Questions being formed

Figure 28 - Steps in Question Generation

AUTOMATED ESSAY GRADING:

DATASET

We obtained the dataset from the link:
https://www.kaggle.com/c/asap-aes/data

The dataset contains 8 essay sets generated from a single prompt. Out of sets 1-8, we have used 1,2,7,8 to train our model because these were narrative essays.
Selected essays range from an average length of 150 to 550 words per response.

Each of these files contains the following columns:

- essay_id: A unique identifier for each individual student essay
- essay_set: 1-8, an id for each set of essays
- essay: The ascii text of a student's response
- rater1_domain1: Rater 1's domain 1 score; all essays have this
- rater2_domain1: Rater 2's domain 1 score; all essays have this
- rater3_domain1: Rater 3's domain 1 score; only some essays in set 8 have this
- domain1_score: Resolved score between the raters; all essays have this
- rater1_domain2: Rater 1's domain 2 score; only essays in set 2 have this
- rater2_domain2: Rater 2's domain 2 score; only essays in set 2 have this
- domain2_score: Resolved score between the raters; only essays in set 2 have this
- rater1_trait1 score - rater3_trait6 score: trait scores for sets 7-8

The training data is provided in a Microsoft Excel 2010 spreadsheet format.

essay_id	essay_set	essay	rater1_dom	rater2_dom	rater3_dom	domain1_s	rater1_dom	rater2
1	1	Dear local newspaper, I think effects computers have on people are great learning skills/affects because they give us time to chat with friends/new people, helps us learn about the globe(astronomy) and	4	4		8		
2	1	Dear @CAPS1 @CAPS2, I believe that using computers will benefit us in many ways like talking and becoming friends will others through websites like facebook and mysace. Using computers can help u	5	4		9		
3	1	Dear, @CAPS1 @CAPS2 @CAPS3 More and more people use computers, but not everyone agrees that this benefits society. Those who support advances in technology believe that computers have a pos	4	3		7		
4	1	Dear Local Newspaper, @CAPS1 I have found that many experts say that computers do not benifit our society. In some cases this is true but in most cases studdies show that computers can help people.	5	5		10		
5	1	Dear @LOCATION1, I know having computers has a positive effect on people. The computers connect families, contain information which is great for peoples education, and are very conveint. Computc	4	4		8		
6	1	Dear @LOCATION1, I think that computers have a negative affect on us! How many people have acess to a camputer daily in america.. @NUM1 and how many people go on at least an hour a @NUM2. Th	4	4		8		
7	1	Did you know that more and more people these days are depending on computers for their safety, natural education, and their social life? In my opinion, the increasing use of computers is not benefitin	5	5		10		
8	1	@PERCENT1 of people agree that computers make life less complicated. I also agree with this. Using computers teaches hand-eye coordination, gives people the ability to learn about faraway places and	5	5		10		
9	1	Dear reader, @ORGANIZATION1 has had a dramatic effect on human life. It has changed the way we do almost everything today. The most well know, is the computer. This device has allowed people do t	4	5		9		
10	1	In the @LOCATION1 we have the technology of a computer. Some say that the computers are good for the society. I disagree, I believe that it is bad for a few reasons. Some of the reasons are obesity, crar	5	4		9		
11	1	Dear @LOCATION1, @CAPS1 people acknowledge the great advances that computers give us, but they also put us at @CAPS1 disadvantages. Computers have been known to help develop health issues. /	4	4		8		
12	1	Dear @CAPS1 @CAPS2 I feel that computers do take away from peoples life and aren't as important than the other factors of life. First of all you know that the world is becoming obease because of lack c	4	4		8		
13	1	Dear local newspaper I raed ur argument on the computers and I think they are a positive effect on people. The first reson I think they are a good effect is because you can do so much with them like if yo	4	3		7		
14	1	My three detaileds for this news paper article is one state you opinion about the effects of computers. Seconde give detailed resons that will persuade of the local newspaper to agree with yor postition.	3	3		6		
15	1	Dear, In this world today we should have everyone useing computers. Computers have a positive effect on people. They teach hand-eye coordination, give people the ability to learn about faraway place	3	3		6		
16	1	Dear @ORGANIZATION1, The computer blinked to life and an image of a blonde haired girl filled the screen. It was easy to find out how life was in @LOCATION2, thanks to the actual @CAPS1 girl explain	6	6		12		
17	1	Dear Local Newspaper, I belive that computers have a negative effect on peoples lives. I belive this because who spend to much time on the computer don't get out as much as they should, don't spend e	4	4		8		
18	1	Dear Local Newspaper, I must admit that the experts are centainly right cause I know people that stay on the computer all day and that's really not good if you ask me. Instead of being on computers lear	4	4		8		
19	1	I aegre waf the evansmant ov tnachnolage. The evansmant ov tnachnolge is being to halp fined a kohar froi alnsas. Tnanchnolage waf ont ot we wod not go to the moon. Tnachnologe evans as we maech	2	2		4		
20	1	Well computers can be a good or a bad thing. I don'@CAPS1 realy see @CAPS2 computers can be a bad thing for me. I also know @CAPS2 computers can or will help people all around the world. I think	3	3		6		
21	1	Dear @CAPS1 of the @CAPS2 @CAPS3 daily, I am writing you to address the issue of computers in our society today. Each day new discoverys are made from the people that use computers in our societ	4	4		8		
22	1	Dear local Newspaper @CAPS1 a take all your computer and given to the people around the world for the can stay in their houses chating with their family and friend. Computers help people around th	2	1		3		
23	1	Dear local newspaper, @CAPS1 you ever see a child on the computer for hours and nothing could get them off? Well I believe that this is very harmful for the child since it doesn't allow them to exersice	5	5		10		
24	1	Dear local newspaper, I've heard that not many people think computers benefit society. I disagree with that. Computers benefit society by teaching hand-eye coordination, allowing people to learn abo	6	5		11		
25	1	Dear @CAPS1, @CAPS2 off, I believe that computers are very helpfull to many people by looking up information, or talking to friends. Although many kids should be spending their time outside, the cor	4	4		8		
26	1	Do you think that computers are useless? Or do you think they could be useful in many ways? I remember @CAPS4 my older sister had to do some research on @CAPS1, the computer gave her all of the i	5	4		9		
27	1	Computers a good because you can get information, you can play games, you can get pictures, But when you on the computer you might find something or someone that is bad or is viris. If ther is a vris y	2	2		4		
28	1	Dear Newspaper, Computers are high tec and have expanded in an everyday thing. They play a big role in society. Computers have been improved to do many different things without them we would ba	5	4		9		
29	1	Dear local newspaper, @CAPS1 people throughout the world use, or own computers. Although there are @CAPS1 people who think computers are good for you, others could say different. While you a	5	4		9		
30	1	Dear Newspaper People, I think that computers do benefit society for a few reasons. Computers make work easier they can do things people can't like solve difficult problems, and kids like playing games	4	4		8		
31	1	I agree that computers deffinately are an advantage to our society. I think this because they help us communicate and video chat with family and friends online, used as a great tool with school work, or	4	6		10		
32	1	Dear Local Newspaper, @CAPS1 name is @PERSON1 and I am a @ORGANIZATION1 citizen. I believe that computers do NOT benefit society of all in @CAPS1 opinion, they make our town worse. People	5	5		10		
33	1	Dear, @ORGANIZATION1 I think the effects that computers do on people are really positive. Computers can be used for all sorts of things. Examples like finding things out about history. People that char	3	3		6		
34	1	Dear @LOCATION1, @CAPS1 is a rumor going around saying that computers are a negitive affect in a childs life. I disagree. A computer can help a kid in so many ways, it can help students with homewor	4	4		8		
35	1	Dear @CAPS1 @CAPS2, I think computers have a positive effect on people. Where would we be without them? Computers teach hand-eye coordination, give people the ability to learn about any subjec	4	5		9		
36	1	Dear @CAPS1 @CAPS2, Have you ever wondered what effect computers have on people? It is a very negative effect. Computers are one of the main reasons of why kids do not exercise and become obese	5	5		10		
37	1	Dear @ORGANIZATION1, @CAPS1 has been brought to my attention that some people feel that computers are bad for us. Some people say that they are a distraction to our physicaland mental health. A	6	6		12		
38	1	Dear local Newspaper, @CAPS1 in the society we live in, more and more people are using computer including me. The computor in my opinion has to be one of the best inventions ever. The bad part is t	4	4		8		
39	1	Dear local Newspaper A lot more people uses computers daily but not everyone agrees that it benefits society. Those people who supports advances in technology believe that computers have a positive	5	5		10		
40	1	Dear local newspaper, my name is @PERSON1 I am @NUM1 and I go to naylor school. I am writing to you convince that computers have a positive effect on people. Please read the following reasons. Co	4	3		7		
41	1	I think computers are good because you can talk to your friends and family on the computers. People needs computers to look for a job. Some people spend to much time on the computers then on hon	1	1		2		
42	1	Dear @CAPS1 @CAPS2 @CAPS3: Computers have a negitive affect on people and their to connect with others because they always are on the computer and never outside. My first reason is that comput	4	4		8		
43	1	The effect of people using computers isn't anything bad couse you learn a lot from it. In some ways when you use the computer and you type, the typing exercises your fingers. Your fingers will get muscl	3	3		6		
44	1	People are spending too much time on computers. People that spend to much time on the computer do not have time to interact with family or friends. People spened to much time on the computer.	4	4		8		
45	1	Dear @LOCATION1, @CAPS1 you think the computers don't help you with research? Well, if you really think about it computers @CAPS1 help you. You can't always gain wait by just looking or staying c	4	4		8		

Fig 29. Essay dataset

DATA ANALYSIS

The dataset comprises 5875 essays. The following graph depicts the number of essays belonging in different scores categories in the range of 0-10.

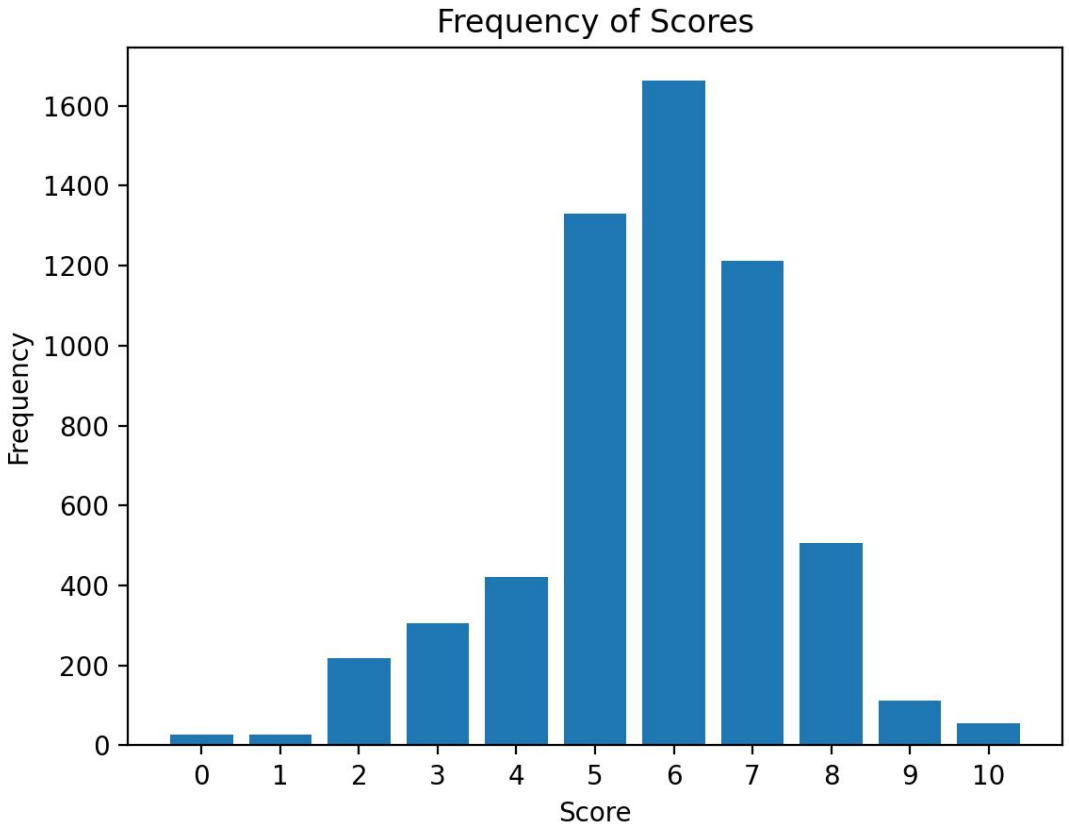


Fig 30. Frequency of scores in the dataset

DATA PREPROCESSING

Features Extraction: Our model grades essays after training on the below mentioned features-

1. Sentence Count:
   The total number of sentences is considered for grading to check if the essay answer is compliant with the word limit specified by the professor. We tokenized the text into sentences using the sent_tokenize function provided by python's nltk library.

2. Word Count:
   The total number of words is considered for grading to check if the essay answer is compliant with the word limit specified by the professor. We tokenized the text into words by using a regex and the nltk library's regexp tokenizer.

3. Spelling Errors:
   The model is trained to grade the essay in accordance with the number of spelling errors (eg. committee spelled as commitee) it has. The python library used for counting the Spelling errors was “SpellChecker”

4. Grammar Errors:
   The model is trained to grade the essay in accordance with the number of grammatical errors(eg who, whom interchanged) it has. We used the ATD (After The Deadline) API for calculating the number of grammatical errors.

5. Topic Coherence Score:
   The topic coherence score specifies the degree to which words in the essay are related to the topic. We used nltk’s wordnet to calculate synsets of words used in topic and essay and then calculated the topic coherence score using the path similarity.

6. Average sentence length:
   It specifies the average length of the sentences written in the essay. We tokenized the text into sentences using the nltk library’s sent_tokenize function and then calculated the average of lengths.

7. Unique Word Count
   The model is trained to determine the number of unique words used by students - a method to test their vocabulary. To count the number of unique words, we tokenized the text into words and counted the number of unique words in the obtained tokens using python set.

8. Flesch Reading Ease:
   It is a measure for calculating the approximate reading level of English-language content. Number of words and length of the words are considered for calculating the Flesch readability score.
   We calculated the flesch reading score using python’s textstat library.

Flesch Reading Ease Score	Readability Level / Category	Education Level
0-29	Very Confusing	College Graduates
30-49	Difficult	College
50-59	Fairly Difficult	High School Senior
60-69	Standard	13 to 15 year-olds
70-79	Fairly Easy	12 year-olds
80-89	Easy	11 year-olds
90-100	Very Easy	10 year-olds

Fig 31. Flesch Reading ease[11]

9. Coleman liau index

   It is another readability test that looks at the difficulty of the text to comprehend. It expresses it as the grade a student in the USA would be able to read it. It is easier to calculate but it is said to be less accurate as it calculates the number of characters in words.

   *Coleman-Liau Index = (5.89 * characters/words) – (0.3 * sentences /(100 * words)) – 15.8*

   We calculated the index using python's textstat library.

Apart from these, the specified(expected) word length of the essay is fed with the training set as well.

The various features are represented in the figure below:

L15 fx

	A	B	C	D	E	F	G	H	I
1	0	1	2	3	4	5	6	7	8
2	sentenceCou	wordCount	spellingError	grammarErr	topicCoherer	avgSentLen	uniqWordCo	fleschReadin	CLIndex
3	16	350	13	1	6005.91943	21.875	173	74.02	8.54
4	20	423	23	3	7228.90636	21.15	205	67.08	7.95
5	14	283	9	9	4965.80535	20.2142857	160	68.2	8.3
6	27	530	48	8	8284.62119	19.6296296	260	53.34	11.26
7	30	473	13	3	8065.4193	15.7666667	210	72.66	8
8	15	247	16	1	3856.13934	16.4666667	135	77.67	6.33
9	30	508	10	0	8264.30686	16.9333333	231	70.94	8.29
10	39	508	10	8	9271.68557	13.025641	223	74.39	8.05
11	35	451	8	4	7716.31849	12.8857143	224	75.2	6.66
12	26	519	8	5	8644.83136	19.9615385	220	75.64	6.62
13	22	330	10	4	5769.50034	15	214	64.3	10.84
14	25	401	31	6	6818.76703	16.04	166	71.85	8.29
15	6	204	16	3	3160.61636	34	122	62.35	5.99
16	25	307	16	8	4735.79458	12.28	134	83.86	6.13
17	13	177	7	2	3269.14867	13.6153846	100	72.16	8.58
18	35	534	28	3	9326.00676	15.2571429	260	63.7	10.55
19	18	347	8	3	5760.93196	19.2777778	149	77.98	7.31
20	15	374	8	13	6644.23523	24.9333333	176	72.09	7.49
21	7	66	26	0	519.146346	9.42857143	53	87.31	6.52
22	11	160	5	4	3283.51918	14.5454546	97	73.98	7.71
23	20	368	13	3	6151.6533	18.4	191	68.81	8.7
24	2	56	2	0	973.653528	28	38	59.98	9.87
25	30	530	23	4	8111.97378	17.6666667	235	70.13	7.89
26	39	576	20	1	10157.9563	14.7692308	261	71.75	8.7
27	16	296	5	3	4798.58	18.5	134	69.82	9.4
28	22	363	17	1	5808.23573	16.5	174	66.88	8.12
29	7	122	10	1	2027.87684	17.4285714	70	76.56	7.43
30	28	363	12	1	6247.42933	12.9642857	195	74.39	8.52
31	23	377	12	3	5865.06199	16.3913044	172	71.24	7.89
32	15	264	3	1	4893.87062	17.6	124	69.72	8.36
33	34	484	19	2	8123.46347	14.2352941	239	71.85	7.36
34	36	499	12	2	8607.81105	13.8611111	237	73.27	7.82
35	12	167	9	3	3266.21955	13.9166667	104	74.29	7.36
36	26	370	15	7	6787.82537	14.2307692	174	81.53	5.56
37	33	421	6	6	7458.07687	12.7575758	181	75.71	7.18

Fig 32. Features of essay dataset

Apart from feature selection, the following preprocessing was done on the dataset:

- Convert the .xlsx to csv using the python library "xlrd"
- Normalise score in the range of 0-10
- Normalise values of all features between 0 and 1

The dataset is split into 80-20 ratio for training and testing respectively.

TECHNIQUE

1. SVM

1.1 EXPLANATION

An SVM algorithm works by finding a hyperplane in a space of dimension m, where m depicts the no. of features which classify the data points distinctly.

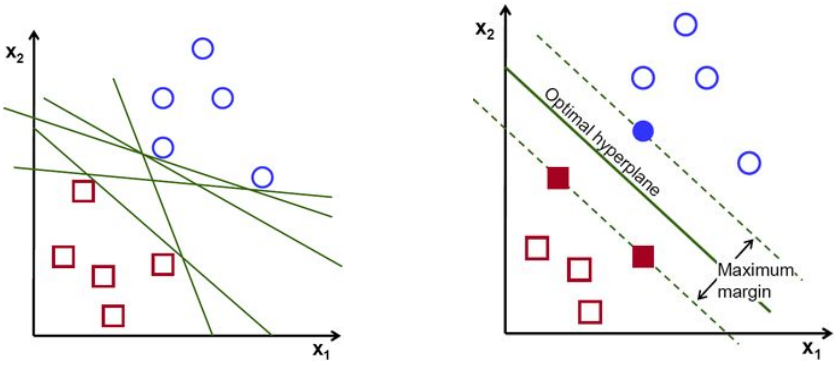


Fig 33. SVM[12]

A hyperplane can be defined as a decision boundary that classifies the data point i.e points on any side of it could be classified into 2 classes.

Number of input Features	Hyperplane
2	Line
3	2-D Plane

Table 3. SVM Hyperplanes

1.2 WHY WE CHOSE THIS ALGORITHM

- SVM takes less computational power and time.
- It is easy to implement.
- It is considered to give more accurate results as compared to other ML algorithms like Linear regression etc.
- Since a linear plane is not sufficient to characterize an essay into the various grades, a gaussian kernel should prove helpful.

1.3 OUR IMPLEMENTATION

Kernel used: Gaussian rbf.

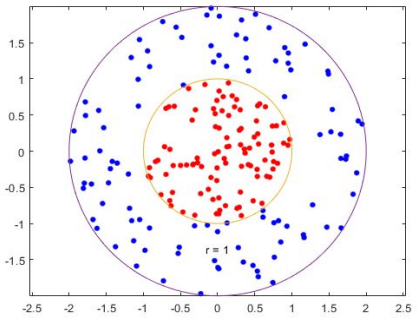


Fig 34. Gaussian kernel[13]

The model is classified and trained on 11 classes with essay scores from 0-10. The steps for training the model are as follows:

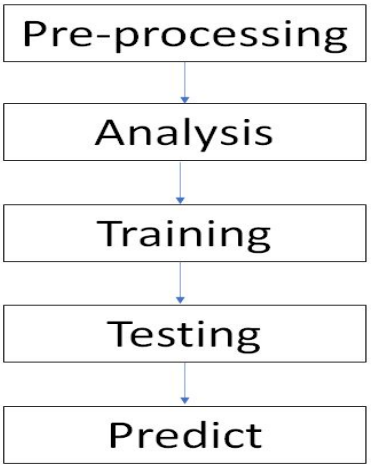


Fig 35. Steps in essay grading- SVM

2. Feed Forward Neural network

2.1 EXPLANATION

A neural network is a Deep learning technique developed to mimic the working of a human neuron in order to intelligently learn patterns. FFNN is named so because the data flows from one layer of neurons to the next in the forward direction with no feedback connections.

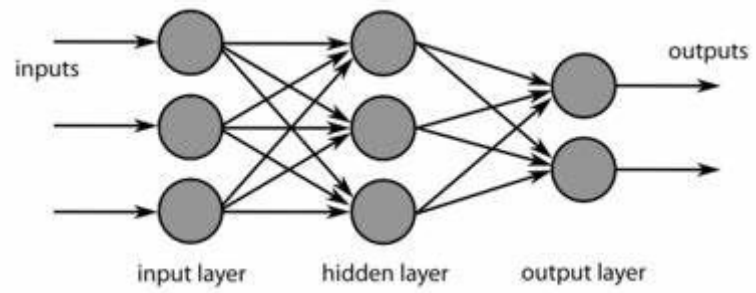


Fig36. FFNN[14]

2.2 WHY WE CHOSE THIS ALGORITHM

- Since the machine learning algorithm did not give a good accuracy, we implemented a DL model as the next step.
- Since the features are already extracted from the essay, and the model is not required to remember word sequences, advanced DL models like LSTM etc that consume a lot of computational time and resources were not suitable for the problem, hence a FFNN has been implemented.

2.3 OUR IMPLEMENTATION

Our model's summary is depicted in the figure below:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 200)               2200
_________________________________________________________________
dense_1 (Dense)              (None, 200)               40200
_________________________________________________________________
dense_2 (Dense)              (None, 200)               40200
_________________________________________________________________
dense_3 (Dense)              (None, 200)               40200
_________________________________________________________________
dense_4 (Dense)              (None, 11)                2211
=================================================================
Total params: 125,011
Trainable params: 125,011
Non-trainable params: 0
_________________________________________________________________
```

Fig 37. Model summary

The hyperparameters used were as follows:

HYPERPARAMETER	VALUE
Number of layers	5
Number of nodes	200, 200, 200,200, 11
Activation functions	ReLu, tanh,Softmax
Optimizer	Adam
Loss function	sparse_categorical_crossentropy
Number of epochs	1000
Batch size	16

Table 4. Hyperparameters in FFNN

The Steps to train the model are as follows:

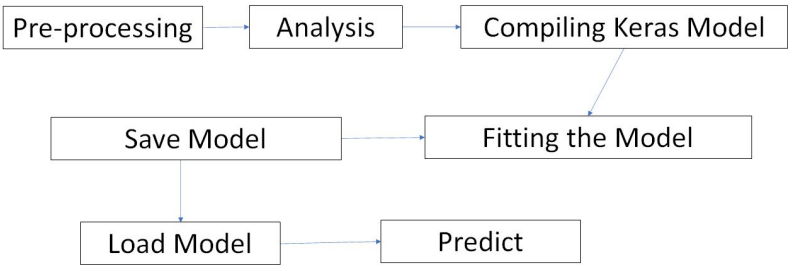


Fig 38. Steps in essay grading - FFNN

INTERNAL PLAGIARISM DETECTION:

DATASET

The Internal Plagiarism model was tested on manually created 25 documents with 4 levels of Internal Plagiarism :

- **Near copy:** The text from the source document was copied into this folder.

- **Light revision**: The documents were constructed by altering the text in the source document. Changes include using synonyms of certain words or changing the grammatical structure with no changes in the order of information found in sentences.

- **Heavy revision:** The dataset has been constructed by rephrasing the text in the source documents. The newly constructed documents have the same meaning as the original document but have been represented using different words and changed structure. The changes include breaking of sentences into 2 or more individual sentences, combining them into a single sentence.

- **Non-Internal Plagiarism** : Completely new documents have been written on the same topic with no relation with the source document.

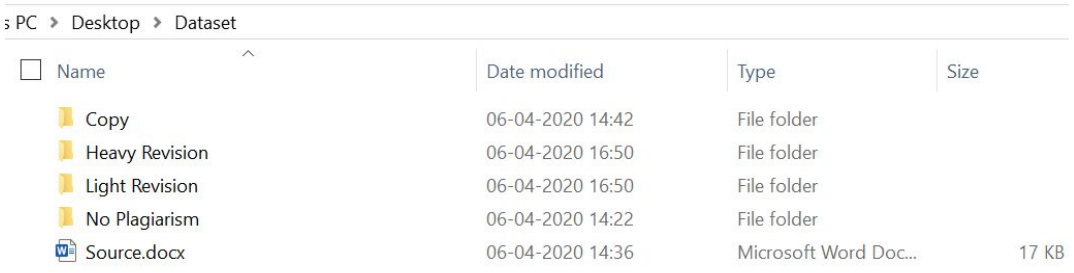


Fig 39. Internal Plagiarism dataset

TECHNIQUES:

We implemented the following techniques in Python to detect Internal Plagiarism in the essays:

1. **Sequence Matcher**

   1.1. EXPLANATION

   SequenceMatcher is primarily used for comparing pairs of input sentences & is a class available in a python module named **"difflib".** *It works by finding the* **longest common subsequence *(LCS)*** *which has zero* **"junk"** *elements.*

   1.2 WHY WE CHOSE THIS ALGORITHM

   We implemented a sequence matcher for plagiarism detection because it is a simple algorithm that is easy to implement and provides one important advantage that changing the order of sentences doesn't decrease the

plagiarism score because it analyses the tokens extracted and does not depend on the order they were obtained in.

1.3 OUR APPROACH

Step 1: Read content from input files

Step 2: Tokenize the essay into words using nltk's regexp tokenizer

Step 3: Compare using SequenceMatcher class

Step 4: Output similarity ratio

```
prigup@Priyas-MacBook-Pro plagiarism_detection % python seqmatcher.py
Contents of File1:  Abraham Lincoln was an American statesman and lawyer who served as the 16th president of th
e United States (1861–1865). Lincoln led the nation through its greatest moral, constitutional, and political c
risis in the American Civil War.[3][4] He preserved the Union, abolished slavery, strengthened the federal gove
rnment, and modernized the U.S. economy. Lincoln was born in poverty in a log cabin and was raised on the front
ier primarily in Indiana. He was self-educated and became a lawyer, Whig Party leader, Illinois state legislato
r, and U.S. Congressman from Illinois. In 1849 he returned to his law practice but became vexed by the opening
of additional lands to slavery as a result of the Kansas–Nebraska Act. He reentered politics in 1854, becoming
a leader in the new Republican Party and he reached a national audience in the 1858 debates against Stephen Dou
glas. Lincoln ran for President in 1860, sweeping the North in victory. Pro-slavery elements in the South equat
ed his success with the North's rejection of their right to practice slavery, and southern states began secedin
g from the union. To secure its independence, the new Confederate States of America fired on Fort Sumter, a U.S
. fort in the South, and Lincoln called up forces to suppress the rebellion and restore the Union.

Contents of File2:  Abraham Lincoln was an American statesman and lawyer who served as the 16th president of th
e United States (1861–1865). Lincoln led the nation with its greatest moral, constitutional, and political cris
is in the American Civil War.[3][4] He reentered politics in 1854, becoming a leader in the new Republican Part
y and he reached a national audience in the 1858 debates against Stephen Douglas. Lincoln ran for President in
1860, sweeping the North in victory. Pro-slavery elements in the South equated his victory with the North's rej
ection of their right to practice slavery, and southern states began seceding from the union. To secure its ind
ependence, the new Confederate States of America fired on Fort Sumter, a U.S. fort in the South, and Lincoln ca
lled up forces to suppress the rebellion and restore the Union. He preserved the Union, abolished slavery, stre
ngthened the federal government, and modernized the U.S. economy. Lincoln was born in poverty in a log cabin an
d was raised on the frontier primarily in Indiana. He was self-educated and turned out to be a lawyer, Whig Par
ty leader, Illinois state legislator, and U.S. Congressman from Illinois. In 1849 Abraham returned to his law p
ractice but became vexed by the beginning of additional lands to slavery as a result of the Kansas–Nebraska Act
.

Similairty ratio:  0.4481288488867835
```

Fig 40. Sequence matcher output

2. Vector Space Model

Vector space model or term vector model is an algebraic model for representing text documents (and any objects, in general) as vectors of identifiers, such as, for example, index terms. It is used in information filtering, information retrieval, indexing and relevancy rankings. Its first use was in the SMART Information Retrieval System.

Documents and queries are represented as vectors.

$$d_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$$

$$q = (w_{1,q}, w_{2,q}, \ldots, w_{n,q})$$

Every dim has a corresponding term. When a particular word appears in the document, its value in the vector is non-zero.

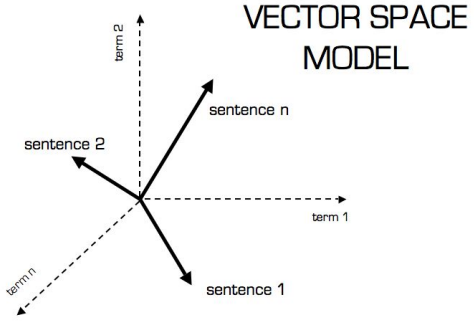


Fig41. Vector space model [15]

When a term is considered as a word, the vector dimensionality is calculated as the number of distinct words in the vocabulary.

2.1 Using Cosine Similarity

2.1.1 EXPLANATION

We first implemented the vector space model based on the cosine similarity. It calculates the cosine of the angle between the two vectors(in our case, two documents).

This metric is as follows:

$$\vec{a} \cdot \vec{b} = \|\vec{a}\| \|\vec{b}\| \cos\theta$$

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \|\vec{b}\|}$$

It will output a value that signifies the magnitude of similarity of documents by looking at the angle (instead of magnitude). It is also majorly used for topic modelling on a similar basis.

2.1.2 WHY WE CHOSE THIS ALGORITHM

Since Sequence matcher uses Longest common subsequence approach, it skips non-matching nodes and hence tolerates a lot of noise. So in some cases, it might give a high similarity score even if the essay was not plagiarised. However, the vector space model overcomes this limitation by representing words as vectors and not skipping any characters or words while calculating the similarity.

2.1.3 OUR APPROACH

Step 1: Read content from input file
Step 2: Tokenize text into words
Step 3: Remove stopwords
Step 4: Compute Document Frequency of words
Step 5: Compute term freq of all words
Step 6: Compute Inverse doc freq (IDF)
Step 7: Compute Term frequency - Inverse Document Frequency(TF-IDF) weight vectors
Step 8: Compute dot product of TF-IDF vectors
Step 9: Compute magnitude of vectors
Step 10: Compute and print cosine similarity

These steps have been diagrammatically explained in the following flowchart:

Fig42. Vector Space Model using Cosine Similarity Flowchart

2.2 Using Jaccard Similarity

2.2.1 EXPLANATION

It measures similarity between the two documents. The value is between 0 and 1. 0 show that documents are dissimilar and 1 show those documents are identical with each other. Values between 0 and 1 show the probability of similarity between the documents.

Jaccard formulation as shown below:

$$J(A,B) = (|A \cap B|) / (|A \cup B|)$$

2.2.2 WHY WE CHOSE THIS ALGORITHM

Since the cosine similarity is focused on topic modelling, two students' submission on the same topic will always result in plagiarism (because of the same topic, cosine similarity will consider the documents to be highly similar

and give a plagiarism score of >=90%). Hence Jaccard similarity which depends on the Bag of words and not on the topic has been implemented.

2.2.3 OUR APPROACH

Step 1: Read content from input file
Step 2: Tokenize text into words
STep 3: Find unique words
Step 4: Remove stopwords
Step 5: Lemmatize the words
Step 6: Find intersection of set of obtained words
Step 7: Find union of set of obtained words
Step 8: Compute and print jaccard similarity

These steps have been diagrammatically explained in the following flowchart:

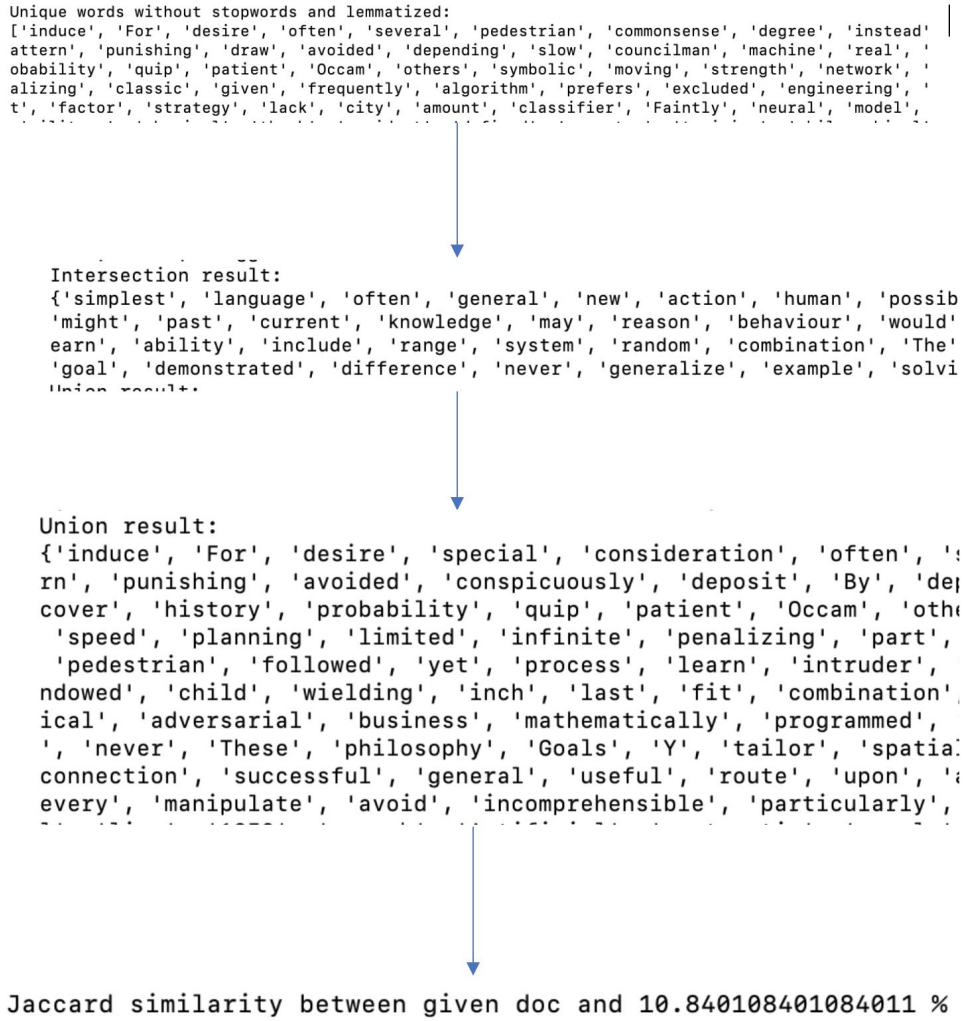


Fig 43. Vector Space Model using Jaccard Similarity Flowchart

TESTING DOCUMENTS

SUMMARY GENERATION ANALYSIS:

Precision and Recall in the Context of ROUGE(Recall-Oriented Understudy for Gisting Evaluation):

"Recall = number of overlapping words / total words in reference summary"

"Precision = number of overlapping words / total words in system summary"

METRIC	TF IDF	TextRank	LSA
Precision	0.75	0.70	0.80
Recall	0.72	0.75	0.78
F1 Score	0.8	0.76	0.8

Table 5 - Results for Text Summarization

QUESTION GENERATION ANALYSIS:

The results are showed in the table below followed by inferences

S. NO	NO OF SENTENCES	NO OF CORRECT QUESTIONS	NO OF INCORRECT QUESTIONS	NO OF QUESTIONS BY HUMAN
1	1	2	0	2
2	1	2	0	2
3	2	3	1	2
4	1	2	0	2
5	1	2	0	2
6	1	2	0	2
7	2	1	1	3
8	1	2	0	2
9	1	2	0	3
10	1	3	0	3

Total no of sentences = 12
Total no of correct questions = 21
Total no of incorrect questions = 2
Total no of questions by the human = 23
The above result shows that the system is working fairly correct with an accuracy of over 90% which can be further improved.

Fig 44. Results of Question generation

ESSAY GRADING ANALYSIS:

METRIC	SVM	Feed Forward Neural Network
Accuracy	0.75	0.96
Precision	0.75	0.8
Recall	0.75	0.8
F1	0.75	0.8
QWK	0.44	0.78

Table 6. Essay grading result

Confusion Matrix

1. SVM

	Actual Positive	***Actual Negative***
Predicted Positive	887	288
Predicted Negative	288	11462

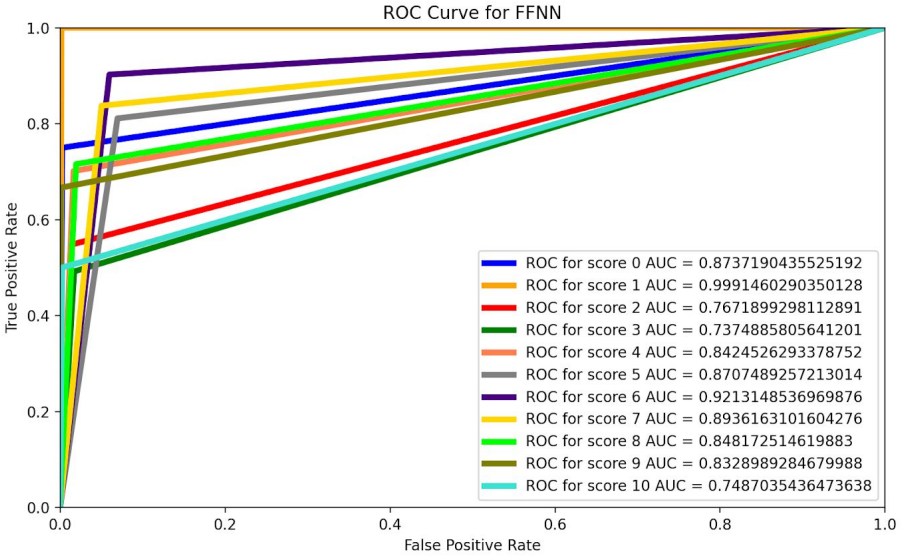
Table 7. Confusion matrix - SVM

2. Feed Forward Neural Network

	Actual Positive	***Actual Negative***
Predicted Positive	947	228
Predicted Negative	228	11522

Table 8. Confusion matrix - FFNN

ROC curves



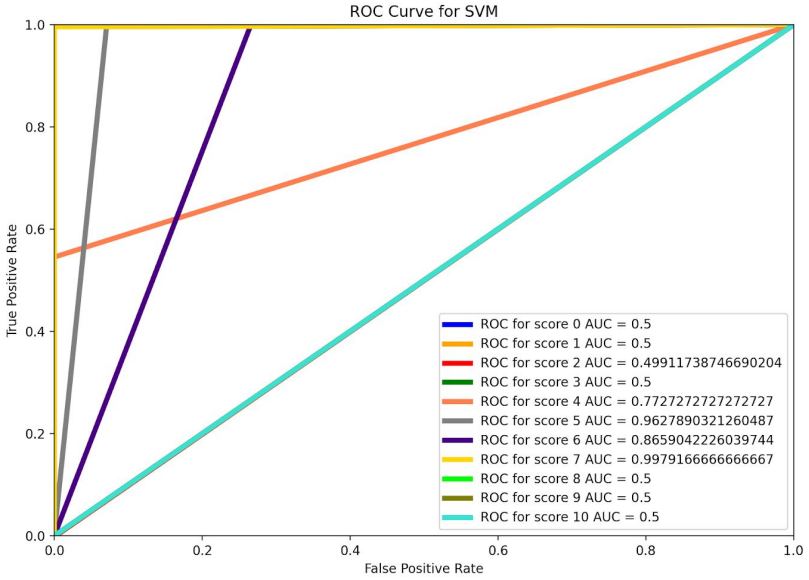


Fig 45,46. ROC - Essay grading

Inference from ROCs:

In the case of SVM, more than half of the classes (score: 0,1,3 etc.) have their ROCs coinciding with the diagonal (AUC = 0.5), which implies that there is no class separation for these classes. However, in the case of FFNN, all the classes have ROCs in the upper half (AUC > 0.5) with some AUCs very close to 1. This implies a better performance and hence we conclude that FFNN is more suitable for the classification purpose.

INTERNAL PLAGIARISM DETECTION ANALYSIS:

File1	File2	Predicted Plagiarism percentage			Actual Plagiarism percentage
		Seqmatcher	Jaccard	Cosine	
Source.docx	Copy1 - D1.docx	100	100	99.99	100
Source.docx	Copy2 - D2.docx	100	100	99.99	100
Source.docx	Copy3 - D3.docx	100	100	99.99	100
Source.docx	Copy4 - D4.docx	100	100	99.99	100
Source.docx	Copy5 - D5.docx	100	100	99.99	100
Source.docx	Light Revision - Para12.docx	92.87	93.45	99.94	92
Source.docx	Light Revision - Para34.docx	97.07	96.33	99.96	96
Source.docx	Light Revision - Para56.docx	92.89	93.29	99.94	93
Source.docx	Light Revision - Para78.docx	98.74	97.53	99.97	98
Source.docx	Light Revision - Para910.docx	84.53	94.44	99.96	94
Source.docx	Heavy revision - Para12.docx	86.91	93.28	99.93	91
Source.docx	Heavy revision - Para34.docx	86.59	96.50	99.96	95
Source.docx	Heavy revision - Para56.docx	75.82	93.12	99.94	91
Source.docx	Heavy revision - Para78.docx	79.02	97.35	99.97	97
Source.docx	Heavy revision - Para910.docx	80.68	94.44	99.96	95
Source.docx	No plag - Doc1.docx	0.66	10.21	93.3	0
Source.docx	No plag - Doc2.docx	0.8	10.84	92.3	0
Source.docx	No plag - Doc3.docx	0.9	11.93	92.9	0

Source.docx	No plag - Doc4.docx	1.5	10.72	92.8	0
Source.docx	No plag - Doc5.docx	1.47	10.86	93.8	0

Table 9. Internal Plagiarism detection observations

The actual values have been computed using an online Internal Plagiarism Checker tool: https://www.prepostseo.com/Internal Plagiarism -comparison-search

Metrics	**Sequence Matcher**	**Vector Space Model**	
		Jaccard	**Cosine**
Mean Absolute Error	3.87	3.21	26.13
Mean Squared Error	47.23	30.6	2182.44
Accuracy	1	1	0.75
Precision	1	1	0.75
Recall	1	1	1
F1 score	1	1	0.85
AUC	1	1	0.5

TABLE10. Internal Plagiarism Detection result

Confusion Matrix

1. **Sequence Matcher**

	Actual Plagiarised	***Actual non plagiarised***
Predicted Plagiarised	5	0
Predicted non-plagiarised	0	15

Table 11. Confusion matrix - Seqmatcher

2. **Vector Space Model using Jaccard Similarity**

	Actual Plagiarised	***Actual non plagiarised***
Predicted Plagiarised	5	0
Predicted non-plagiarised	0	15

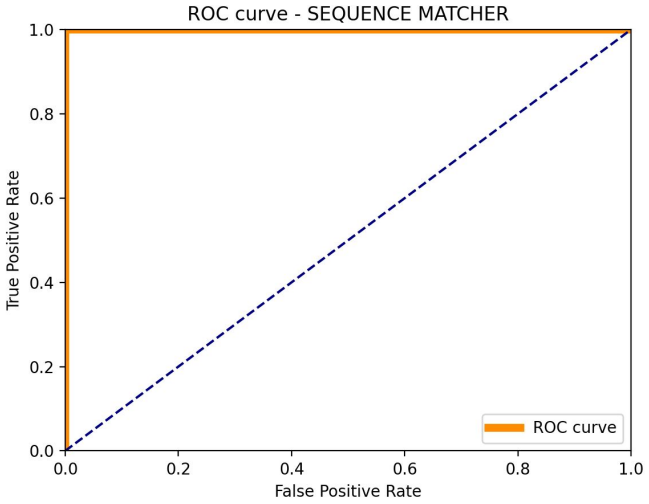
Table 12. Confusion matrix - Jaccard

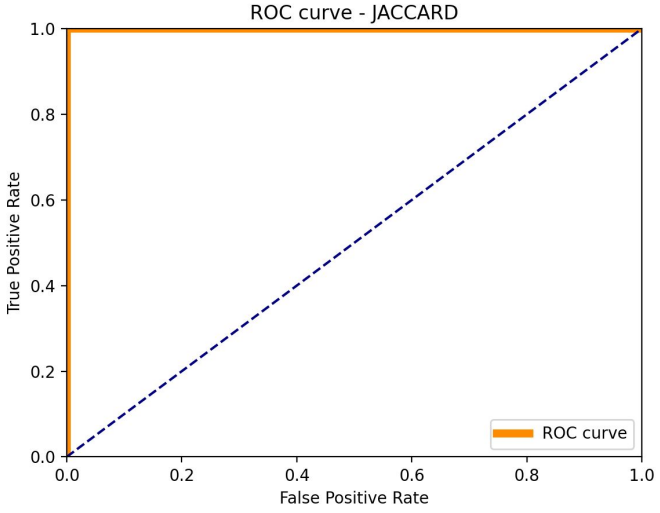
3. **Vector Space Model using Cosine Similarity**

	Actual Plagiarised	***Actual non plagiarised***
Predicted Plagiarised	0	5
Predicted non-plagiarised	0	15

Table 13. Confusion matrix - Cosine

ROC curves





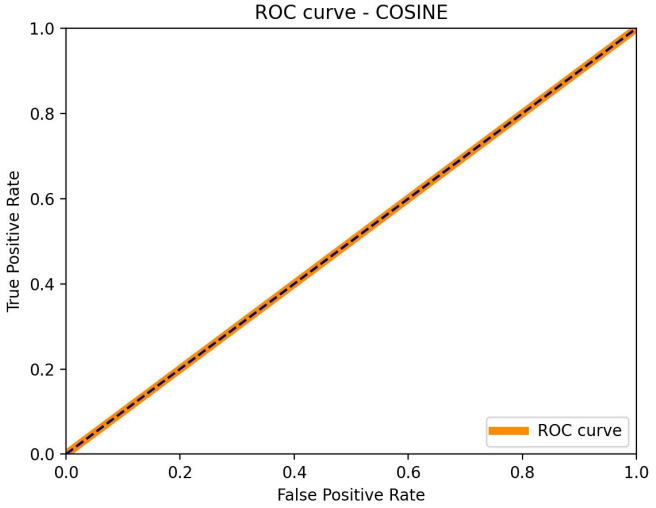


Fig 47,48,49- ROC Curves - Internal Plagiarism

Inference from ROCs:

The ROC for model based on cosine similarity coincides with the diagonal (AUC = 0.5), which implies that it has no evident class separation for the two classes, viz., Plagiarised and Non-plagiarised. This is due to the fact that cosine similarity is based on topic modeling so it considers two documents to be plagiarised if they are based on the same topic. The ROC curves for jaccard similarity and sequence matcher lie in the upper half of the plane and have an AUC value of 1. This implies that, for the given test cases, both performed equally well and differentiated between both the classes correctly. Further metrics like MSE etc were then used to choose one out of the two.

STUDENT FEEDBACK

We conducted a survey among our batchmates to understand the general perception of our product. The answers have been depicted graphically as follows:

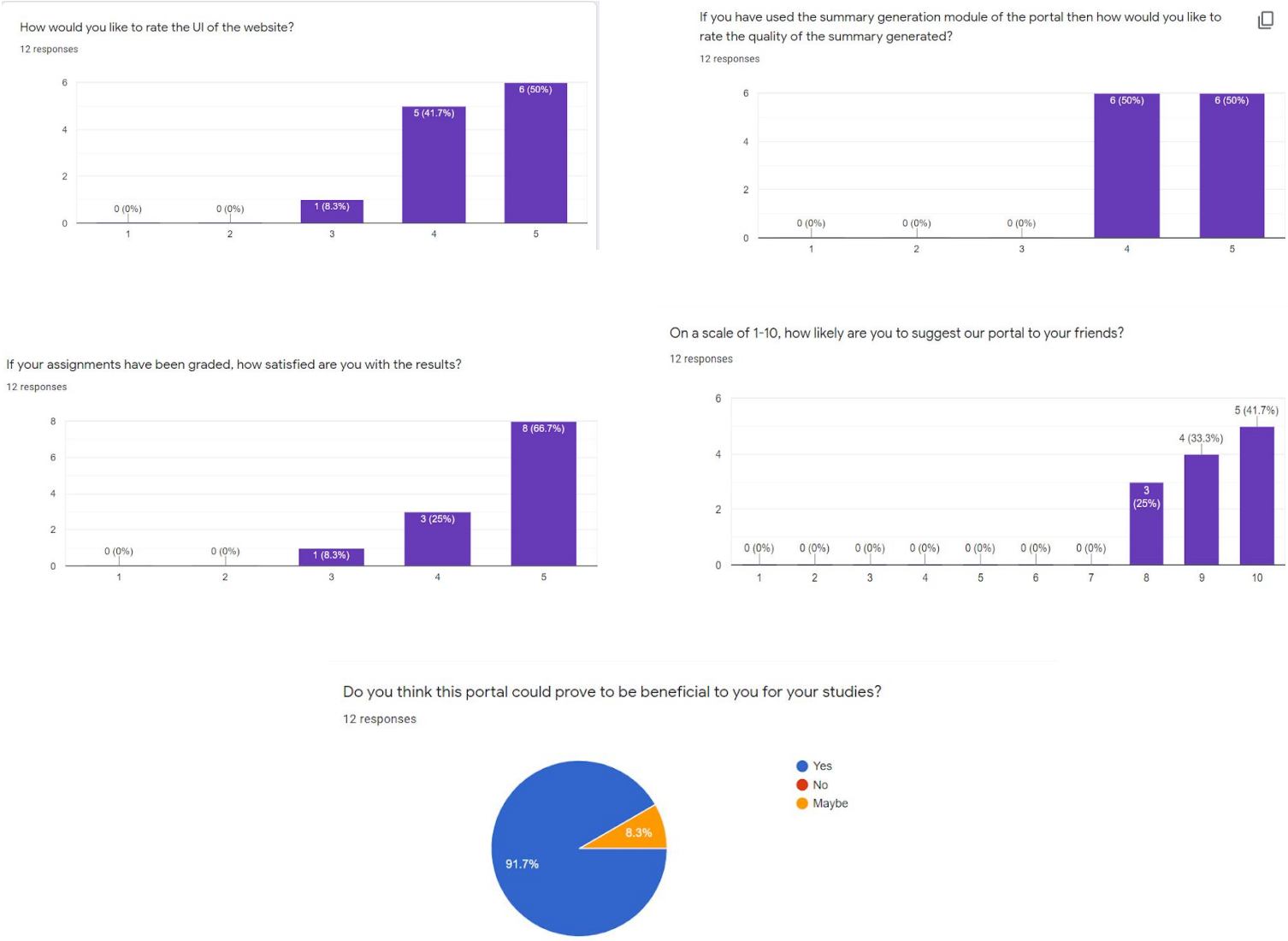


Fig 50. Student Feedback

INSTALLATION GUIDELINES

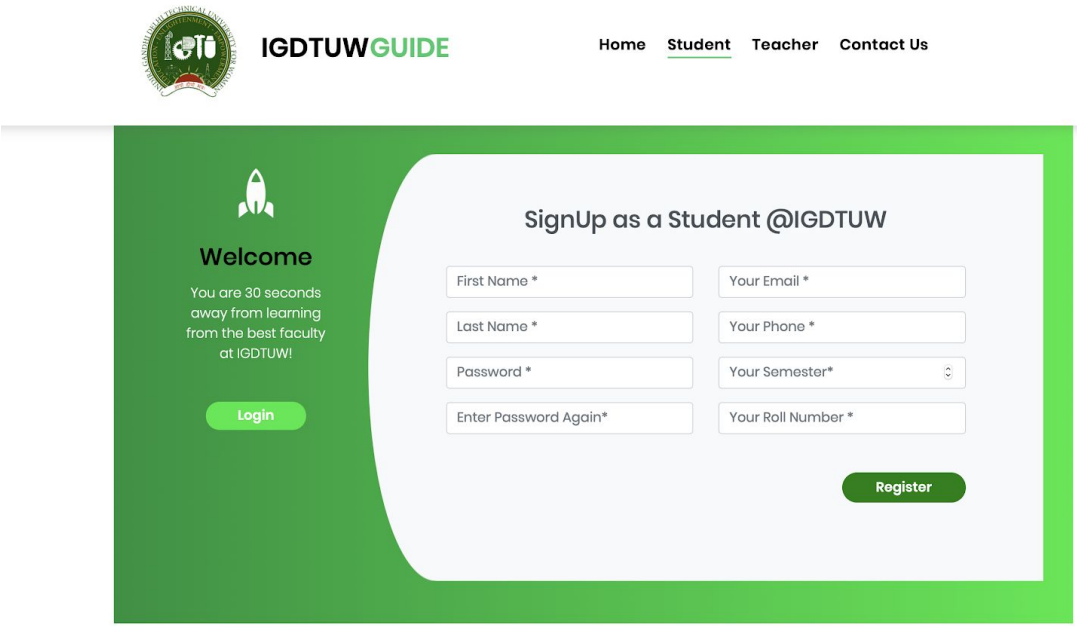


Fig 51. Login for student

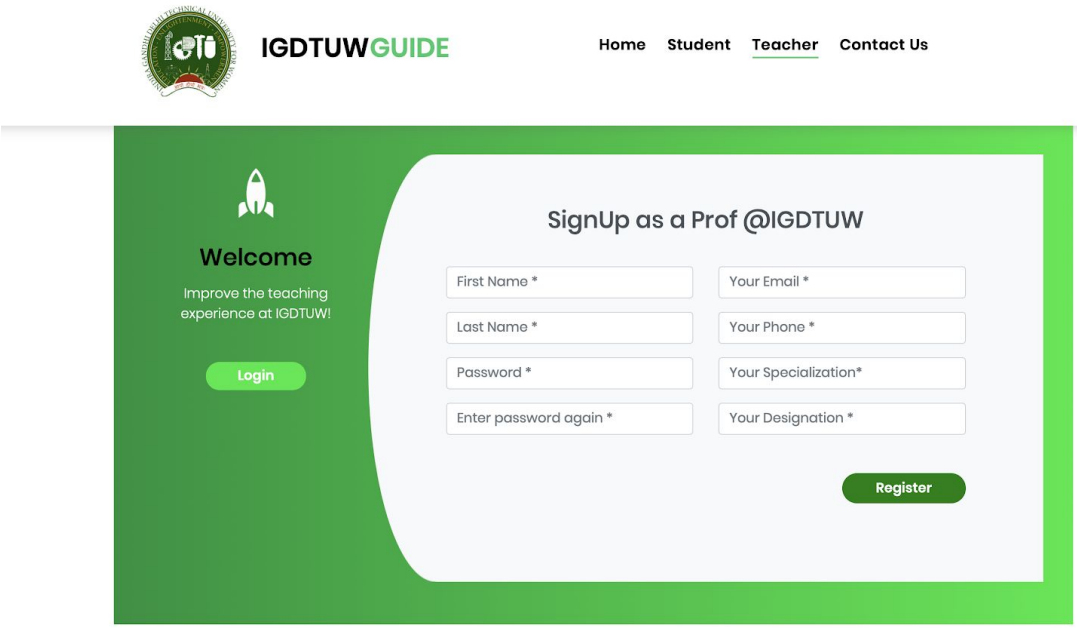


Fig 52. Login for Professor

CONCLUSION

Based on the results of the experiments performed, LSA has been chosen to implement the summary generation module due to the following reasons:

1. It reduces the dimensionality of the original **text**-based dataset.
2. It helps us understand what each topic is encoding.
3. Analyzes word association in **text** corpus.
4. Finds relations between terms.

For essay grading, feed forward had a better accuracy and QWK score than SVM, so the former has been used in our deployed portal.

For Internal Plagiarism Detection, the "*Vector Space Model using Cosine similarity"* produced the lowest accuracy. This observation was due to the fact that cosine similarity of documents is based on the semantics of the sentence, i.e. their meanings and not just on the bag of words. Hence, it considers two documents on the same topic to be highly similar and therefore, it was not found suitable for detecting Internal Plagiarism . On the other hand, the other two techniques are based on the words used and not their meaning. Out of "*Sequence Matcher"* and "*Vector space model using Jaccard similarity",* the latter produced a smaller error rate and hence is the chosen model for our portal.

The following models are found to give the best performance metrics:

1. Summary Generation Analysis - LSA
2. Question Generation Analysis - NLTK Python Library
3. Grading Analysis - Feed Forward Neural Network
4. Internal Plagiarism Detection analysis - Vector Space Model using Jaccard Similarity

Hence, on the basis of our analysis, following are the metrics of the final proposed model:

Metrics	**Summary - LSA**	**Grading - Feed Forward Neural Network**	**Internal Plagiarism - Vector Space Model using Jaccard**
Precision	0.80	0.8	1
Recall	0.78	0.8	1
F1 Score	0.8	0.8	1
Accuracy	NA	0.96	1

Table 14. Results of the final model

FUTURE SCOPE

The future enhancement of this application is to process the various types of documents by the creation of a mobile application for the text summarization approach in smart phones. Taking into consideration different languages other than English e.g. Spanish, French, Hindi for grading, Internal Plagiarism detection, question and summary generation. Addition of more features such as assignment grade predictor so that students can work better according to the predictions and also prompting areas of improvement to the student for his overall enhancement. Generating questions by doing web searching for keywords. Checking if the essay is copied from the web (External Plagiarism). Taking into account the veracity of facts stated in the essays while automatically grading them. Checking the Logical Coherence of sentences in the essay while automatically grading it. Creating summaries directly by taking URLs of the web instead of providing a text document.

REFERENCES

[1] K. Khadilkar, S. Kulkarni and P. Bone, "Plagiarism Detection Using Semantic Knowledge Graphs," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-6, doi: 10.1109/ICCUBEA.2018.8697404.

[2] E. Hunt et al., "Machine Learning Models for Paraphrase Identification and its Applications on Plagiarism Detection," 2019 IEEE International Conference on Big Knowledge (ICBK), Beijing, China, 2019, pp. 97-104, doi: 10.1109/ICBK.2019.00021.

[3] Chitra, A., and Anupriya Rajkumar. "Internal Plagiarism detection using machine learning-based paraphrase recognizer." *Journal of Intelligent Systems* 25.3 (2016): 351-359.

[4] Moiyadi, Hamza Shabbir, et al. "NLP Based text summarization using semantic analysis." *International Journal of Advanced Engineering, Management and Science* 2.10 (2016).

[5] K. Shetty and J. S. Kallimani, "Automatic extractive text summarization using K-means clustering," 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), Mysuru, 2017, pp. 1-9, doi: 10.1109/ICEECCOT.2017.8284627.

[6] Gupta, Som, and S. K. Gupta. "Deep Learning in Automatic Text Summarization." *International Journal of Computer Science and Information Security (IJCSIS)* 16.11 (2018).

[7] Jethwani, Himanshu, Mohd Shahid Husain, and Mohd Akbar. "Automatic Question Generation from Text." *International Journal for Innovations in Engineering, Science and Management Available online at: www.ijiesm.com Volume 3, Issue 4, April 2015*

[8] Zhao, Siyuan, et al. "A memory-augmented neural model for automated grading." *Proceedings of the Fourth (2017) ACM Conference on Learning@ Scale*. 2017.

[9] M. Monjurul Islam and A. S. M. Latiful Hoque, "Automated essay scoring using Generalized Latent Semantic Analysis," *2010 13th International Conference on Computer and Information Technology (ICCIT)*, Dhaka, 2010, pp. 358-363, doi: 10.1109/ICCITECHN.2010.5723884.

[10] Taghipour, Kaveh, and Hwee Tou Ng. "A neural approach to automated essay scoring." *Proceedings of the 2016 conference on empirical methods in natural language processing. 2016.*

REFERENCES FOR FIGURES

[1]https://tse2.mm.bing.net/th?id=OIP.TSSNRg5ScoRuHlPpaPhj4QHaE3&pid=Api&P=0&w=241&h=159

[2] https://www.onlineparaphrase.net/wp-content/uploads/2017/01/plagiarism-statistics.png

[3]https://image.slidesharecdn.com/factsaboutplagiarismandtoolstoavoidit-160310060648/95/tools-to-avoid-plagiarism-and-some-valuable-facts-about-plagiarism-6-638.jpg?cb=1457590277

[4] https://towardsdatascience.com/comparing-text-summarization-techniques-d1e2e465584e

[5]https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarization-textrank-python/

[6]http://www.ijaerd.com/papers/finished_papers/Automatic%20Question%20Generation%20from%20Paragraph-IJAERDV03I1213514.pdf

[7] https://github.com/indrajithi/genquest

[8] -https://github.com/indrajithi/genquest

[9] https://freesoft.dev/program/134517569

[10] https://github.com/dipta1010/Automatic-Question-Generator

[11]https://tse3.mm.bing.net/th?id=OIP.pv2RCKuPNOyRz3y9wlVVkAHaDZ&pid=Api&P=0&w=421&h=193

[12] https://dimensionless.in/introduction-to-svm/

[13] https://i.stack.imgur.com/7yM2K.png

[14]https://upload.wikimedia.org/wikipedia/commons/thumb/c/c2/MultiLayerNeuralNetworkBigger_english.png/381px-MultiLayerNeuralNetworkBigger_english.png

[15]https://tse2.mm.bing.net/th?id=OIP.E3GrXGeqgW3GZ03_WzpO5AHaFj&pid=Api&P=0&w=215&h=162

ANNEXURE

PLAGIARISM REPORT

Submission date: 23-May-2020 10:11PM (UTC+0530)
Submission ID: 1330533761
File name: Major_project_report_Priya.pdf (4.47M)
Word count: 8718
Character count: 46414

v3

ORIGINALITY REPORT

13%	10%	7%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

UNDERTAKING REGARDING ANTI-PLAGIARISM

We hereby, declare that the material/ content presented in the report is free from plagiarism and is properly cited and written in our own words. In case, plagiarism is detected at any stage, we shall be solely responsible for it. A copy of the Plagiarism Report is also enclosed.

Jahnvi Tyagi
(00601012016)

UNDERTAKING REGARDING ANTI-PLAGIARISM

We hereby, declare that the material/ content presented in the report is free from plagiarism and is properly cited and written in our own words. In case, plagiarism is detected at any stage, we shall be solely responsible for it. A copy of the Plagiarism Report is also enclosed.

Priya Gupta
(01701012016)

UNDERTAKING REGARDING ANTI-PLAGIARISM

We hereby, declare that the material/ content presented in the report is free from plagiarism and is properly cited and written in our own words. In case, plagiarism is detected at any stage, we shall be solely responsible for it. A copy of the Plagiarism Report is also enclosed.

Mehar Kaur
(06201012016)